An Evaluation Methodology and Framework for Semantic Web Services Technology

Dissertation

zur Erlangung des akademischen Grades
Doktor-Ingenieur (Dr.-Ing.)

vorgelegt dem Rat der Fakultät für Mathematik und Informatik
der Friedrich-Schiller-Universität Jena

von Diplom-Informatiker Ulrich Küster
geboren am 08.03.1980 in Wuppertal

Gutachter

1. **Prof. Dr. Birgitta König-Ries**
   Friedrich-Schiller-Universität Jena, D-07743 Jena
2. **Prof. Dr. Thomas Kirste**
   Universität Rostock, D-18059 Rostock
3. **Prof. Dr. Manfred Hauswirth**
   National University of Ireland, Galway, Irland

Tag der öffentlichen Verteidigung: 18. Juni 2010

Ehrenwörtliche Erklärung

Hiermit erkläre ich,

- dass mir die Promotionsordnung der Fakultät bekannt ist,
- dass ich die Dissertation selbst angefertigt habe, keine Textabschnitte oder Ergebnisse eines Dritten oder eigenen Prüfungsarbeiten ohne Kennzeichnung übernommen und alle von mir benutzten Hilfsmittel, persönliche Mitteilungen und Quellen in meiner Arbeit angegeben habe,
- dass ich die Hilfe eines Promotionsberaters nicht in Anspruch genommen habe und dass Dritte weder unmittelbar noch mittelbar geldwerte Leistungen von mir für Arbeiten erhalten haben, die im Zusammenhang mit dem Inhalt der vorgelegten Dissertation stehen,
- dass ich die Dissertation noch nicht als Prüfungsarbeit für eine staatliche oder andere wissenschaftliche Prüfung eingereicht habe.

Bei der Auswahl und Auswertung des Materials sowie bei der Herstellung des Manuskripts haben mich folgende Personen unterstützt:

- Prof. Dr. Birgitta König-Ries

Ich habe die gleiche, eine in wesentlichen Teilen ähnliche bzw. eine andere Abhandlung bereits bei einer anderen Hochschule als Dissertation eingereicht: Ja / Nein.

Jena, 8. Februar 2010

[Ulrich Küster]

Deutsche Zusammenfassung

Die folgende Zusammenfassung in deutscher Sprache gibt einen kompakten Überblick über die Inhalte dieser Arbeit. Sie orientiert sich an der Gliederung der Dissertation und stellt jeweils Querbezüge zur ausführlichen Behandlung der entsprechenden Themen in der eigentlichen Arbeit her.

Einleitung, Stand der Forschung und Zielsetzung

Die vorliegende Arbeit beschäftigt sich mit der Evaluation von Semantischen Web Diensten (Semantic Web Services, SWS, siehe Kapitel 2.3). Unternehmen organisieren heute ihre IT-Systeme mittels dienstorientierter Architekturen (siehe Kapitel 2.2). Hier werden Ressourcen (etwa die Funktionalität, die ein Rechner bereitstellen kann) als Dienste gekaspelt und über definierte Schnittstellen nach außen angeboten. So aufgebaute Systeme sind potentiell sehr flexibel und dynamisch an Veränderung anpassbar. Eine zentrale Forschungsfragestellung der Informatik ist, wie das Potential dieser Architekturen vollständig ausgeschöpft werden kann, insbesondere inwieweit eine automatische Zusammenstellung von Diensten zum Erreichen einer gewünschten Funktionalität möglich ist (siehe Kapitel 1.1). Semantische Web Dienste versuchen diese Automatisierung mittels maschinenverständlicher Beschreibungen von Diensten zu erreichen (siehe Kapitel 2.1 und 2.3).

In den letzten Jahren wurden eine Vielzahl von Ansätzen in diesem Gebiet entwickelt, die Praxistauglichkeit der Verfahren und ihre relativen Stärken und Schwächen sind jedoch weitgehend unbekannt. Es fehlt an etablierten Methoden, die Ansätze objektiv und zuverlässig zu evaluieren (siehe Kapitel 3.1). Der resultierende Mangel experimenteller Evaluation erweist sich als kritisches Hemmnis für den Forschungsfortschritt und die Übertragung der Forschungsergebnisse in die Wirtschaft (siehe Kapitel 1.2).

Ziel der Arbeit ist es, diese Lücke zu schließen und Methoden zur verlässlichen und aussagekräftigen Evaluation verschiedener Ansätze zur Automatisierung des di-

enstorientierten Rechnens zu entwickeln (Kapitel 1.3). Die Arbeit folgt dem Ansatz, Standardbenchmarks zu entwickeln die im Rahmen gemeinschaftlicher Evaluationsinitiativen in der Forschergemeinde Anwendung finden (siehe Kapitel 1.4). Ein detaillierter Überblick über die Struktur der Arbeit wird in Kapitel 1.5 präsentiert.

Modell zur Evaluation Semantischer Dienste

Um die Grundlage eines systematischen Ansatzes für die Evaluation zu legen, wird zunächst ein konzeptionelles Modell entwickelt, welches die möglichen Kriterien der Evaluation definiert und einen Katalog von Anforderungen an Evaluationen bereitstellt (Kapitel 4).

Die Evaluationskritieren werden mit Hilfe des Goal-Question-Metric Ansatzes aus der Softwaretechnik hergeleitet (Kapitel 4.1). Eine Literaturrecherche gibt einen Überblick über die Ziele, die der Entwicklung semantischer Dienste zu Grunde liegen. Diese Ziele werden dann mit Hilfe von konkreten Fragen operationalisiert. Die konkreten Fragen wiederum ermöglichen die Ableitung von fünf Dimensionen der Evaluation: Performanz und Skalierbarkeit, Benutzerfreundlichkeit und Aufwand, Korrektheit, Unabhängigkeit von Dienstanbietern und -nutzern sowie Funktionsumfang und Automatisierung.

Desweiteren wird ein Anforderungskatalog an Evaluationen semantischer Dienste entwickelt (Kapitel 4.2). Der Anforderungskatalog unterstützt die Qualität von Evaluationen und ermöglicht zudem eine Meta-Evaluation. Er basiert auf Evaluationsstandards der Deutschen Gesellschaft für Evaluation, die unter Berücksichtigung relevanter Arbeiten aus verwandten Gebieten konkretisiert werden.

Das präsentierte Modell zur Evaluation semantischer Dienste ermöglicht eine detaillierte, strukturierte Analyse existierender Arbeiten im Gebiet (Kapitel 4.3 und 4.4). Basierend auf dieser Analyse werden drei offene Probleme ausgewählt, für die im Rahmen der Dissertation Lösungen entwickelt werden (Kapitel 4.5). Diese werden im Folgenden beschrieben.

Test Daten zur Evaluation Semantischer Dienste

Qualitativ hochwertige Testdaten stellen die essentielle Grundlage jeglicher Evaluation dar. Eine Lösung zur Bereitstellung dieser Daten wird in Kapitel 5 präsentiert. Basierend auf einer Anforderungsanalyse werden verfügbare Daten zur Evaluation semantischer Dienste untersucht. Die Untersuchung betrachtet dabei sowohl Daten, die frei im Netz verfügbar sind (Kapitel 5.1), als auch explizit entwickelte Testdatensätze (Kapitel 5.2). Es stellt sich heraus, dass Daten nicht im notwendigen Umfang zur Verfügung stehen. Insbesondere existieren beispielsweise keine nennenswerten

öffentlich verfügbaren Daten für den weit verbreiteten WSMO/WSML Ansatz. Zudem werden die derzeit verfügbaren Daten den Anforderungen hinsichtlich ihrer Qualität und Vielseitigkeit nicht gerecht.

Die Arbeit argumentiert, dass wegen des Aufwandes, qualitativ hochwertige Testdaten im nötigen Umfang zu erstellen und um die gewünschte Vielfalt und Objektivität der Testdaten zu gewährleisten, Standardtestdaten gemeinschaftlich von der Forschergemeinde als Ganzes entwickelt werden sollten. Dies ist jedoch nur umsetzbar, wenn eine verteilte Entwicklung von geeigneten Werkzeugen unterstützt wird.

Zu diesem Zweck wird OPOSSum (Online POrtal for Semantic Services), ein Portal zum Austausch und für die gemeinschaftliche Entwicklung von Testdaten für semantische Dienste entwickelt (Kapitel 5.3). Das Portal dient dem Zweck, die Wiederverwendung und Weiterentwicklung existierender Daten zu unterstützen, ihre Strukturierung, Dokumentation und Nutzbarkeit zu verbessern und eine Nutzung und einen Vergleich über verschiedene Formalismen hinweg zu ermöglichen.

Das Portal wurde als öffentlich zugängliche quelloffene Webanwendung implementiert und auf verschiedenen Konferenzen vorgestellt. Die verbreiteten existierenden Testdatensätze wurden in das Portal integriert und auf diese Art komfortabler zugänglich gemacht. Die Nutzbarkeit des Portals wird durch die Entwicklung des neuen Jenaer Geographie Datensatzes (JGD) illustriert (Kapitel 5.4). Dieser Datensatz verbessert den Stand der Forschung hinsichtlich wesentlicher Anforderungen und findet im Rahmen der Dissertation für erweiterte Evaluationen Verwendung.

Benchmark für den Funktionsumfang von SWS Ansätzen

Im Anschluss an die Arbeiten zu SWS Testdaten wird ein Benchmark für die Erfassung und Zertifizierung des Funktionsumfangs von Ansätzen zur semantischen Dienstsuche, -auswahl und -ausführung vorgestellt (Kapitel 6). Der Benchmark basiert auf einer Sammlung von Problemszenarien die in natürlicher Sprache spezifiziert sind. Jedes Szenario definiert eine Menge konkreter Dienstangebote und strukturiert sich in einzelne Problemstufen (Kapitel 6.4). Diese sind jeweils durch eine Anzahl spezifischer Dienstanfragen repräsentiert. Die Aufgabe besteht darin, den zu einer Anfrage optimal passenden Dienst auszuwählen und ggf. auszuführen.

Jede Problemstufe ist zudem mit einer Menge grundlegender funktionaler Anforderungen verknüpft, welche bewältigt werden müssen, um die Dienstanfragen der entsprechenden Stufe korrekt zu verarbeiten. Ein Katalog entsprechender Anforderungen, wie die Fähigkeit zur Repräsentation von Zahlen, zur Berechnung arithmetischer Ausdrücke, zur Wiedergabe und Auswertung komplexer Nutzerpräferenzen oder zur Nutzung von Informationen welche dynamisch von Web Service Schnittstellen bezogen werden müssen, ist Teil des Benchmarks (Kapitel 6.5).

Die Evaluationsmethodik sieht vor, dass Teilnehmer einer Evaluation mit Hilfe ihrer Technologie Lösungen zu den definierten Problemszenarien entwickeln. Diese Lösungen werden auf Workshops präsentiert. Der Workshop verifiziert korrekt gelöste Problemstufen und zertifiziert darüber die funktionalen Anforderungen, die ein Ansatz korrekt erfüllen kann (Kapitel 6.3). Darüber hinaus sieht der Benchmark vor, dass Teilnahmer Arbeiten verfassen, welche die Unterschiede verschiedener Technologien anhand der konkreten Lösungen zu den Problemszenarien untersuchen und erläutern. Diese Arbeiten werden gemeinschaftlich von den Entwicklern der verglichenen Technologien verfasst, was einen fairen Vergleich sicherstellt und einen wesentlichen Beitrag für ein besseres gegenseitiges Verständnis der Stärken und Schwächen der verglichenen Ansätze leistet. Die Evaluationsmethodik wurde über mehrere Jahre erfolgreich im Rahmen der SWS Challenge Evaluationsinitiative implementiert und ausgeführt. Entsprechende Ergebnisse und Vergleiche werden in Kapitel 6.6 und Anhang B.1 präsentiert.

Benchmark für Semantischen Dienstvergleich

Der zweite Benchmark der im Rahmen der Dissertation entwickelt wurde behandelt die Evaluation semantischer Matchmaker (Kapitel 7). Die Problemstellung besteht darin eine Menge von Diensten gemäß Ihrer Ähnlichkeit mit einem fiktiven Wunschdienst zu sortieren (Kapitel 7.2). Anders als im ersten Benchmark müssen hierbei nicht spezifische und komplexe Dienstanfragen einmalig und vollautomatisch ausgeführt werden, sondern Softwarekomponenten bezüglich Ihrer allgemeinen Verwendbarkeit zur Einbettung in eine Applikation eingeschätzt werden.

Der Benchmark verbessert den Stand der Forschung (Kapitel 7.3) in drei wichtigen Aspekten. Auf praktischer Ebene definiert und verwendet er detailliertere und realistischere Dienste. Auf methodischer Ebene wird ein neuer Evaluationsaufbau präsentiert, der es erstens ermöglicht, Matchmaker formalismenübergreifend zu evaluieren, der zweitens realische Bedingungen emuliert, bei denen die Beschreibungen für Dienstangebote und -anfragen unabhängig voneinander entwickelt werden und der drittens eine Analyse des Einflusses der Verwendung unterschiedlich detaillierter Dienstbeschreibungen ermöglicht (Kapitel 7.4). Auf analytischer Ebene werden neue Evaluationsmetriken definiert, die eine detailliertere, zuverlässigere und feinere Bewertung der Qualität der erzeugten Sortierung ermöglichen.

Zu diesem Zweck werden Aspekte rund um den Begriff der Relevanz im Kontext des semantischen Matchmaking untersucht. Zwei neue Relevanzmodelle, welche die verbreitete binäre Relevanz zu abgestufter und mehrdimensionaler Relevanz erweitern werden vorgestellt (Kapitel 7.5). Desweiteren wird ein Experiment zur Untersuchung der Zuverlässigkeit von Referenzeinschätzungen menschlicher Experten durchgeführt. Das Experiment zeigt eine hohe Inkonsistenz in den üblicherweise ver-

wendeten Referenzeinschätzungen und einen Zusammenhang zwischen dem verwendeten Relevanzmodel und dem Grad an Inkonsistenz. Basierend auf diesen Ergebnissen wird eine Methodik zur Entwicklung zuverlässiger Einschätzungen vorgestellt (Kapitel 7.6).

Desweiteren werden mehrere Evaluationsmaße aus dem Gebiet des Information Retrieval vorgestellt, welche die zusätzlichen Informationen die in nicht-binären Referenzeinschätzungen enthalten sind nutzen. Die Maße werden bezüglich ihrer Zuverlässigkeit untersucht. Dabei werden eine Reihe von Problemen identifiziert und hinsichtlich dieser Probleme verbesserte Maße entwickelt (Kapitel 7.7).

Der Benchmark wurde implementiert und im Rahmen der S3 Contest Evaluationsinitiative ausgeführt (Kapitel 7.8). Basierend auf den Daten aus dieser Evaluation wird die Zuverlässigkeit des Benchmarks detailliert diskutiert. Dabei werden insbesondere die Auswirkungen verschiedener Relevanzmodelle, inkonsistenter Referenzeinschätzungen und verschiedener Evaluationmaße auf die Evaluationsergebnisse untersucht (Kapitel 7.9). Es wird gezeigt, dass binäre Relevanz hochgradig empfindlich für Änderungen im verwendeten Relevanzmodel ist und daher mit Vorsicht verwendet werden sollte. Graduelle Relevanz ist erheblich stabiler und sollte daher bevorzugt werden. Inkonsistente Referenzeinschätzungen scheinen die Evaluationsergebnisse nur geringfügig zu beeinflussen und erweisen sich somit als weniger problematisch als erwartet. Im Gegensatz dazu hat die Wahl des Eavluationsmaßes erheblichen Einfluss auf die Evaluationsergebnisse. Die redundante Verwendung verschiedener Maße wird daher empfohlen. Es wird erwartet, dass diese Ergebnisse erheblich dazu beitragen, zukünftige SWS Matchmaker Evaluation zuverlässiger und aussagekräftiger zu machen.

Validierung

In Kapitel 8 werden die Beiträge der Dissertation validiert. Zunächst wird der Fokus der Arbeit auf gemeinschaftliche Evaluation innerhalb der Forschergemeinde diskutiert. Dies umfasst eine positive Abschätzung, dass das Gebiet des semantischen dienstorientierten Rechnens die Voraussetzungen für die erfolgreiche Durchführung gemeinschaftlicher Evaluationsinitiativen erfüllt. Desweiteren werden die Erfahrungen aus mehreren Jahren Organisationstätigkeit für solche Evaluationsinitiativen wiedergegeben. Die Diskussion kommt zu dem Ergebnis, dass der gemeinschaftliche Ansatz der Arbeit erstrebenswert und zielführend ist, wenn auch der Aufwand für die Organisation oder die Teilnahme an gemeinschaftlichen Evaluationen erheblich ist (Kapitel 8.3).

Das konzeptuelle Modell zur Evaluation semantischer Dienste wird bezüglich seiner Vollständigkeit und Fundiertheit validiert. Die Fundiertheit wird anhand der Anwendbarkeit und korrekten Anwendung der Methode zur Herleitung des Modells

attestiert. Die Vollständigkeit wird gezeigt indem die praktische Anwendbarkeit des Modells zur Diskussion und Abgrenzung der existierenden Evaluationsansätze dargelegt wird (Kapitel 8.4).

Die Beiträge für bessere Testdaten werden evaluiert indem das entwickelte Portal an seinen Zielen gemessen wird. Die Diskussion zeigt, dass das Portal seinen Zielen gerecht wird. Dieser Schluss wird durch die Nützlichkeit des Portals bei der Entwicklung des Jenaer Geographie Datensatzes und durch seine Akzeptanz in der Forschergemeinde zusätzlich unterstützt (Kapitel 8.5).

Schlussendlich werden die zwei Benchmarks die im Rahmen der Dissertation entwickelt wurden evaluiert. Zunächst wird diskutiert, dass die Benchmarks ihren jeweiligen Designzielen gerecht werden. Als zweites werden die Benchmarks formal mit Hilfe der Evaluationsstandards des konzeptionellen Modells bewertet. Darauf folgend wird die Verbreitung der Benchmarks in die Forschergemeinde dargelegt bevor die Stärken und Schwächen der Benchmarks auf einem allgemeineren Niveau diskutiert werden. Es wird gezeigt, dass beide Benchmarks ihre Designziele erreichen und ferner alle Evaluationsstandard erfüllen, die meisten davon zudem ohne Einschränkungen (Kapitel 8.6 und 8.7).

Neben der formalen Evaluation illustriert die erfolgreiche Durchführung der Benchmarks und die positive Aufnahme der Benchmarks bei den Teilnehmern der entsprechenden Evaluierungen die Akzeptanz und damit Qualität der Benchmarks.

Ausblick

Die Arbeit schließt mit einer Zusammenfassung und einer Diskussion weiterführender Forschungsarbeiten (Kapitel 9). Zu den möglichen zukünftigen Arbeitsfeldern gehören inkrementelle Erweiterungen der präsentierten Benchmarks (Kapitel 9.2.1) sowie die Erstellung komplementärer Benchmarks für Evaluationskriterien und Anwendungsfälle die von der vorliegenden Arbeit noch nicht behandelt wurden (Kapitel 9.2.2). Es bleibt zu hoffen, dass die Beiträge der Dissertation helfen, das Verständnis für verschiedene Ansätze des semantischen dienstorientierten Rechnens zu verbessern. Wir hoffen, dass sie Grundlage produktiver Arbeiten über die weitere Verbesserung der Evaluationsmethodiken, wie auch der evaluierten Technologien selbst wird, um die weitere wissenschaftliche Entwicklung des Feldes zu unterstützen.

Acknowledgments

Many people have accompanied me during my Ph.D. work over the last five years. This thesis would not have been possible without their invaluable support and contributions. I would like to thank all of them for helping me and playing a role in shaping this thesis.

My thanks go first and foremost to Prof. Dr. Birgitta König-Ries who supervised my thesis. During all of my Ph.D. work she served as a continuous source of encouragement. Her general activity, energy and openness to new ideas were truly inspiring. She managed to motivate when motivation was needed, to push when progress was necessary and to provide support and cover when things were difficult. Above all, she was always immediately available when I sought advice. I sincerely admire her ability to provide this level of support despite of her numerous activities and responsibilities and am very grateful for the excellent and productive working environment she created.

I'm also thankful to my colleagues who contributed to this great environment. Without them, working in our group would have been much less fun. I discussed many aspects of my thesis with them, in particular with Fedor Bakalov. Fedor has also been an endless source of competent help with graphics and layout issues. Fedor's pursuit of excellence and perfection were inspiring and very motivating. Mohammed Hamdy, with whom I shared an office, was the best office mate I could imagine. His sense of humor and occasional self-mockery help make him an amazing person to be around. I would like to particularly thank him for his outstanding politeness and helpfulness in all things. I should also mention the coffee breaks with Torsten Dettborn and, later, Fedor and Mohammed. The breaks were not only a place of regular exchange of ideas; but also a source of motivation, especially during the critical phases of my thesis writing.

Prof. Dr. Thomas Kirste and Prof. Dr. Manfred Hauswirth served as external reviewers of my dissertation. I would like to thank them for their helpful comments and all the effort and dedication to reviewing my work. Prof. Dr. Kirste also made me more aware of the aspects of my thesis regarding the philosophy of science

in general and in particular how research communities approach and define new problem areas.

Central contributions of my work are based on the involvement of the scientific community during the definition of reference benchmarks and their execution within community evaluation initiatives. I'm very grateful to the many people who were involved in these activities. My horizons were broadened and scientific work enriched by this community.

Dr. Charles Petrie co-founded the Semantic Web Service Challenge. This initiative and his work there were instrumental in rousing my interest in Semantic Web Service evaluation and community evaluation initiatives. Charles chaired the SWS Challenge for most of the time that I worked as a co-organizer and discovery chair of the challenge. He shared his great experience in many scientific discussions with me. I'm very thankful for everything I learned through him.

Prof. Dr. Matthias Klusch initiated the S3 Contest, the other evaluation initiative of central importance to my thesis. In the beginning, there was an unfortunate miscommunication between us. However, instead of getting upset, Matthias called me to clarify the issue. Out of this call, a very fruitful cooperation developed. I sincerely appreciate Matthias' frank and honest way of dealing with conflict. Without his call, our cooperation might not have happened. I would also like to acknowledge the invaluable support that he and his colleague Patrick Kapahnke gave me while I was organizing the S3 Contest Cross Evaluation Track in 2009.

Besides Charles and Matthias, several other people also co-organized the evaluation initiatives I was involved in. It was a pleasure to work with them and I would like to acknowledge their contributions and the insights they shared with me. Among others I would like to particularly thank Holger Lausen, Prof. Dr. Tiziana Margaria, Christian Kubczak, Dr. Michal Zaremba, Srdjan Komazec, Dr. Federico Facca and Prof. Emanuele Della Valle.

Organizing any community evaluation initiative is meaningless without participants in the evaluation campaign. Many people participated in the evaluation activities I organized or participated myself in and I am very indebted for the energy, time and enthusiasm they invested and for everything I learned from them. It was through these individuals that my work was given an immediate and practical meaning. Besides those I already mentioned, I would particularly like to thank Maciej Gawinecki, Dr. Matteo Palmonari, Dario Cerizza, Andrea Turati, Dr. Maciej Zaremba, Dr. Tomas Vitvar, Michael Maximilien, Dr. Liliana Cabral, Oliver Müller and Dengping Wei.

Last but not least, I'm indebted to my parents and all of my family and friends. They supported me over many, many years and provided help and advice whenever I needed it. Were it not for them, I would not be the person I am today. Finally, I would like to especially thank Bettina for everything she gives to me and for making me very happy during the challenging final phase of my Ph.D.

Abstract

To foster reuse state of the art software engineering has been driven over decades by the trend towards more and more component based software development. In recent years another trend towards more and more distributed and more loosely coupled systems could be observed. Service oriented architectures (*SOAs*) are the latest product of this long-reaching development. Web services in particular have become increasingly popular as the probably most prominent implementation of a SOA. The grand vision of the web service paradigm is to have a rich library of millions of web services available online that provide access to information, functionality or resources of any kind and that can be easily integrated into existing applications or composed in a workflow-like fashion to form new applications.

Even though this promising technology has already proven to be an effective way of creating widely distributed and loosely coupled systems, the manual tasks of integrating the services is still labor intensive and thus expensive work. Thus — following the vision of the Semantic Web [BLHL01] — the idea of Semantic Web Services (SWS) was introduced [MSZ01], applying the principles of the semantic web to the web service paradigm. Numerous efforts providing formal semantic descriptions for component services have been put forward. Based on such descriptions, frameworks are designed to support automated or semi-automated dynamic service discovery, composition, binding and invocation, enabling the creation of new kinds of flexible and adaptable applications and reducing long-term development cost.

SWS related research has flourished in recent years and the presented approaches become increasingly more sophisticated and mature. Yet, very little effort is put into the evaluation of the various approaches. Until very recently there were no comparative evaluations and it was impossible to find two systems which had been evaluated on the same use cases. Existing evaluations were mostly concentrated either on artificially synthesized datasets under questionable assumptions or based on one or two use cases or case studies which are often reverse engineered from the solution. This shortcoming hinders future scientific progress and the transfer of research results into industry.

The presented thesis argues that established evaluation methodologies and standard benchmarks that allow efficient comparative evaluations of the competing approaches are needed for the further advancement of the field. It further argues that community based benchmarking initiatives are the most suitable vehicle to define such standards. Common initiatives not only promote the relevance, quality and acceptance of evaluations, their existence often also results in greater communication and collaboration among different researchers leading to a stronger consensus on the community's research goals.

To lay the foundation for thorough SWS evaluations, the important questions what to evaluate, which criteria to use, how to measure those criteria and how to achieve reliability, validity and impartiality need to be answered. Thus, as the first major contribution of this thesis, a comprehensive and well-founded conceptual model for SWS technology evaluation that identifies the criteria of evaluation and requirement standards to ensure and promote evaluation quality is presented. Based upon this model, the state of the art is reviewed in detail. Three further contributions towards improved SWS technology evaluation are motivated and provided.

First, issues around test data for SWS evaluation are investigated. Based upon a requirements analysis, deficiencies of the state of the art are discussed. It is argued that future data should be developed collaboratively. To support this, a portal is designed and implemented, which allows sharing and reusing test data more effectively and creating better datasets in a distributed way. The portal's utility is illustrated by using it for the development of an exemplary new test collection that improves the state of the art in important aspects.

Second, a methodology and benchmark for evaluating the functional scope of SWS frameworks is developed. It allows certifying capabilities of different approaches in an objective way and additionally establishes an understanding of the fundamental challenges in the covered area. The methodology has been implemented and executed under the umbrella of the SWS Challenge community evaluation initiative.

Third, a methodology and benchmark for evaluating SWS matchmakers is developed. It assesses the correctness of SWS discovery across formalisms based upon a set of improved measures. Additionally, it provides means for investigating other important aspects like the necessary coupling between service requesters and providers or the effects of more or less comprehensive service descriptions. It has been implemented and executed under the umbrella of the S3 Contest, the second community initiative in the area.

The thesis contributions are validated by discussing them methodologically and assessing them with respect to the requirements catalogue provided as part of this thesis. Additional validation results from their successful dissemination to the community via the before mentioned initiatives, which is also documented within this thesis. The thesis concludes with a summary and an outlook on future research in the area.

Contents

List of Tables

List of Figures

Part I.

Foundation

CHAPTER 1

Introduction

> If you can not measure it, you can not improve it.
>
> *(Lord Kelvin)*

This chapter starts with a general overview of the thesis and its context. The relevance of the thesis problem and the chosen solution approach will be further motivated in Section 1.2. Based upon this motivation, the thesis objectives will be defined in Section 1.3 followed by a more detailed description of the research contributions and solution approach in Section 1.4. The chapter concludes with an outline of the structure of the thesis in Section 1.5.

1.1. Overview

This thesis is situated at the convergence of two major trends present in nowadays computer science.

For one thing, state of the art software engineering has been driven over decades by the trend towards more and more component based software development. This has been complemented in recent years by another trend towards more and more distributed and more loosely coupled systems. Service oriented architectures (*SOAs*) are the latest product of these long-reaching developments. Web services in particular have become increasingly popular and are currently the most prominent implementation of a SOA. The grand vision of the Web service paradigm is to have a rich library of tens of thousands Web services available that provide access to information, functionality or resources of any kind and that can be easily integrated into existing applications or composed in a workflow-like fashion to form new applications.

For another thing, computers have evolved from few isolated mainframes to a network of a myriad ubiquitous computing devices. The World Wide Web in particular has thriven and grown to become "the universe of network-accessible information, the embodiment of human knowledge"[1]. Yet, the Web was designed to deliver content to humans. With its staggering growth in size and complexity, its full potential can only be unfolded, if computers are enabled to meaningfully manipulate its data and process its semantics. Thus, the Semantic Web was proposed as an extension of the current one, in which information is given well-defined meaning, better enabling computers and people to work in cooperation [BLHL01].

While SOAs and Web services have already proven to be an effective way of creating widely distributed and loosely coupled systems, the location and integration of the services is still labor intensive and expensive work. Thus the idea of Semantic Web Services (*SWS*) was introduced [MSZ01], applying the principles of the Semantic Web to the SOA/Web service paradigm. SWS intend to facilitate the automation of mediation, choreography and discovery for Web Services using semantic annotations, thus making the vision of seamless automated integration of data and processes on a global Web scale reality.

SWS related research has thriven and attracted a huge amount of effort and funding. Within the sixth EU framework program (which ran from 2002 to 2006) alone at least 20 projects with a combined funding of more than 70 million Euro dealt directly with Semantic Web Services[2]. The current seventh EU framework still funds another 16 projects in the area[3]. A search for the term "Semantic Web Services" at Google Scholar yields 11.500 results, a staggering amount of research for a time frame of less than a decade[4].

This gives a good impression of the importance being currently put on this field of research. The huge amount of effort spent into SWS research has resulted in numerous proposals of ontology based semantic descriptions for component services [Klu08b]. Based on such descriptions, a plethora of increasingly sophisticated techniques and algorithms for the automated or semi-automated dynamic discovery, composition, binding, and invocation of services have been proposed [Klu08a].

Given the variety of technologies proposed and their increasing complexity, meaningful comparisons of different approaches based upon standard problems following established methodologies and procedures become key to the further advancement of the technologies. However, surveys have shown that surprisingly little effort has been spent towards the comparative evaluation of the competing approaches [KLKR07, KZ08, HKZ08, KKRPK08]. Until recently there were no comparative evaluations and it was impossible to find two systems which had been

[1] `http://www.w3.org/WWW/`
[2] `http://cordis.europa.eu/fp6/projects.htm`
[3] `http://cordis.europa.eu/fp7/projects_en.html`
[4] As of December 2009: `http://scholar.google.com`

evaluated on the same use cases. Evaluations were mostly concentrated either on artificially synthesized datasets under questionable assumptions or based on one or two use cases for which it was not clear, whether they were reverse engineered from the solution.

In other words: "There are many claims for such technologies in academic workshops and conferences. However, there is no scientific method of comparing the actual functionalities claimed. [...] Progress in scientific development and in industrial adoption is thereby hindered" [LPZ07].

There are striking parallels to this situation in the history of related areas:

> "[in the experiments] ... there have been two missing elements. First [...] there has been no concerted effort by groups to work with the same data, use the same evaluation techniques, and generally compare results across systems. The importance of this is not to show any system to be superior, but to allow comparison across a very wide variety of techniques, much wider than only one research group would tackle. [...] The second missing element, which has become critical [...] is the lack of a realistically-sized test collection. Evaluation using the small collections currently available may not reflect performance of systems in large [...] and certainly does not demonstrate any proven abilities of these systems to operate in real-world [...] environments. This is a major barrier to the transfer of these laboratory systems into the commercial world." [Har92]

This quote by Donna Harman addressed the situation in text retrieval research prior to the establishment of the series of TREC conferences[5] in 1992 but seems to perfectly describe the current situation in SWS research. Harman continued:

> "The overall goal of the Text REtrieval Conference (TREC) was to address these two missing elements. It is hoped that by providing a very large test collection, and encouraging interaction with other groups in a friendly evaluation forum, a new thrust in information retrieval will occur." [Har92]

From the perspective of today, it is clear that her hope regarding the positive influence of the availability of mature evaluation methods to the progress of information retrieval research was well justified. In fact, "retrieval effectiveness has doubled since TREC began" [Voo05]. This corresponds to a finding of Sim and colleagues who have developed a general theory of benchmarking [SEH03, Sim03]. They observe that the creation and widespread use of a benchmark within a research area is frequently accompanied by rapid technical progress and community building:

> "Creating a benchmark requires a community to examine their understanding of the field, come to an agreement on what are the key problems, and

[5] `http://trec.nist.gov/`

> encapsulate this knowledge in an evaluation. Using the benchmark results in a more rigorous examination of research contributions, and an overall improvement in the tools and techniques being developed. Throughout the benchmarking process, there is greater communication and collaboration among different researchers leading to a stronger consensus on the community's research goals." [SEH03]

This thesis follows these ideas. It is motivated by the belief that an established evaluation methodology and standard benchmarks that allow the comparative evaluation of different frameworks are needed for the advancement of SWS related research.

The development of such benchmarks requires answers to the fundamental research questions related to the evaluation of SWS technology: What are the appropriate criteria for evaluation? How can various fundamentally different SWS approaches be compared effectively? How can such comparison be guaranteed to be unbiased and balanced? Generally, how can the relative advantage of some SWS technology over another one, and ultimately over existing traditional software engineering techniques be reproducibly proven or disproven?

The research objectives and contributions of this thesis are thus twofold:

- Develop a well founded theoretical framework that explores the nature of semantic service evaluation: What to evaluate, which criteria to use and how to achieve validity, reliability and efficiency of the evaluation process.

- Provide reference benchmarks for selected evaluation criteria and use cases that prove the applicability of the theoretical evaluation framework and solve concrete benchmarking needs in the area by establishing procedures how to objectively and effectively compare approaches and results across systems.

Before we define these objectives in more detail in Section 1.3 we will first further motivate the need for benchmarking and experimentation in the area as well as the community-based approach that we followed to pursue this work.

1.2. Motivation

This thesis is based on two main assumptions, which need to be explained, motivated and justified prior to diving into the details of the thesis itself.

First, it is based on the fundamental belief that evaluation and validation of proposed technologies are indispensable for the advancement of computer science in general and SWS related research in particular. Especially comparative evaluations are viewed as essential for scientific progress and experimental validations as essential for industrial adoption of research results. Second, it is based on the idea that

community based evaluation campaigns are the best, if not the only feasible way of performing comparative evaluations and reliable experimental validations.

Both assumptions relate fundamentally to the culture of research in an area. As such, they can not be easily proven or disproven. We will therefore motivate them through historical examples and through references to work explicitly dedicated to the question of how research should be done in computer science.

1.2.1. Experimentation in Computer Science

In some ways, computer science is different from both the natural sciences and the engineering sciences [Har93, Den05]. Being less than a century old, it is much younger than other sciences. It had thus less time to establish a stable culture of how to conduct research than other areas. One of the more objectively observable differences between computer science and other sciences is a notable difference in the number of experimental papers as well as papers with experimental validation being published in the area. Repeated studies, especially by Tichy and colleagues as well as Zelkowitz and colleagues, have found that fewer computer science publications contain evaluations of research contributions and those that do use less rigorous evaluation techniques [TLPH95, HT06, ZW97, ZW98b, Zel06, Zel08].

This issue has repeatedly been acknowledged as slowing down scientific progress in the area. As early as 1976, Turing Award laureates Newell and Simon highlighted the benefits of experimental work in their Turing Award lecture:

> "Neither machines nor programs are black boxes: they are artifacts that have been designed, both hardware and software, and we can open them up and look inside. We can relate their structure to their behavior and draw many lessons from a single experiment." [NS76]

In an article titled "Empirical studies to build a science of computer science", Basili and Zelkowitz also advocate a greater emphasis on experimental work:

> "Empirical evidence sometimes supports and sometimes does not support intuition. When it does, one might feel that empirical evidence is unnecessary. This is fallacious reasoning since the way we build knowledge is through studies, first recognizing relationships (that is, A is more effective than B under the following conditions), then evolving the relationship quantitatively (such as A provides a 20% improvement over B). Demonstrating that gravity exists satisfies our intuition, but being able to measure it adds detail to our understanding. When the evidence does not support our intuition, we must change our mental models and identify hypotheses and conditions for why it doesn't." [BZ07]

Similarly, Tichy discusses the question whether computer scientists should experiment more and concludes:

> "I advocate balance [between theory, engineering and experimentation] [...] because of the following principal benefits:
>
> - Experimentation can help build a reliable base of knowledge and thus reduce uncertainty about which theories, methods, and tools are adequate.
> - Observation and experimentation can lead to new, useful, and unexpected insights and open whole new areas of investigation. Experimentation can push into unknown areas where engineering progresses slowly, if at all.
> - Experimentation can accelerate progress by quickly eliminating fruitless approaches, erroneous assumptions, and fads. It also helps orient engineering and theory in promising directions.
>
> Conversely, when we ignore experimentation and avoid contact with reality, we hamper progress." [Tic98]

As Tichy's quote indicates, the lack of experimentation or, more generally, evaluation is also considered being responsible for hampering adoption of research results to industry, a critical problem for the engineering aspects of computer science. Zelkowitz and Wallace criticize the prevailing research culture in computer science with this respect:

> "Research funding in the computer sciences has been relatively easy to obtain, and building a new technology is easier, and more fun, than showing that such a technology is effective.[...] It is no wonder that the corporate world does not know which technologies to apply since the research world has done such a poor job of explaining its results." [ZW98a]

> "Without a confirming experiment, why should industry select a new method or tool? On what basis should another researcher enhance the language (or extend a method) and develop supporting tools? As a scientific discipline we need to do more than simply say, 'I tried it, and I like it.' " [ZW98b]

Fenton, Pfleeger and Glass put forward a similar critique:

> "As a result, many research findings published can be characterized as 'analytical advocacy research.' That is, the authors describe a new concept in considerable detail, derive its potential benefits analytically, and

> recommend the concept be transferred to practice. Time passes, and other researchers derive similar conclusions from similar analyses. Eventually the consensus among researchers is that the concept has clear benefits. Yet practitioners often seem unenthused. [...] Something important is missing from this picture: rigorous, quantitative experimentation." [FPG94]

Fortunately, a trend towards greater emphasis on experimental work can be observed in recent years. For one thing, the more recent quantitative studies on experimentation in computer science indicate a growing number of papers with experimentally validated claims [HT06, Zel06, Zel08]. For another thing, a number of initiatives drawing attention to the deficit and dedicated to ameliorate the situation have appeared. Among those are a special issue of the Communications of the ACM on Experimental Computer Science [Fei07], a panel on "Performance Evaluation and Experimental Assessment – Conscience or Curse of Database Research?" at VLDB 2007 [MM07], the new SIGMOD "Experimental Repeatability Requirements" [SIG07], the newly established track for "Experiments and Analyses Papers" at VLDB 2008[6] and 2009[7], or the funding of an EU infrastructure project on Semantic Evaluation At Large Scale, which just started in 2009[8].

1.2.2. Community-Based Evaluation Approaches

Having motivated the need for experimentation in computer science, we now turn to discussing how computer scientists validate their claims, if they do. It was already mentioned above that typically less rigorous evaluation techniques are used than in other sciences. Zelkowitz and Wallace classified 612 journal papers according to the data collection method used to validate the claims in the papers [ZW98b]. Another 346 papers were classified in a new study in 2006 [Zel08]. Tichy et al. performed a similar study with similar results on 403 papers in 1995 [TLPH95] and another 133 papers in 2005 [HT06]. Zelkowitz and Wallace summarize their findings:

> "Experimentation is one of those terms frequently used incorrectly in the computer science community. Papers are written that explain some new technology and then 'experiments' are performed to show the technology is effective. In almost all of these cases, this means that the creator of the technology has implemented the technology and shown that it seems to work. Here, 'experiment' really means an example that the technology exists or an existence proof that the technique can be employed. Very

[6] `http://www.cs.auckland.ac.nz/research/conferences/vldb08/index.php/Calls`
[7] `http://vldb2009.org/?q=node/4`
[8] `http://www.seals-project.eu/`

> rarely does it involve any collection of data to show that the technology adheres to some underlying model or theory of software development, or that it is effective, as 'effective' was defined previously, to show that application of that technology leads to a measurable improvement in some relevant attribute." [ZW98b]

This quote emphasizes that in engineering, new methods, new approaches and new technology should result in a measurable improvement of the previous state of affairs. It is ultimately not satisfying and not sufficient to show that a problem, for which a solution already exists, can also be solved in an alternative way, if we fail to also show at least a single advantage of the new approach over the previous ones.

Unfortunately, comparisons with competing approaches are not only extremely labor intensive, they may also be inappropriate and suffer from biases. Such bias results from two problems: insufficient knowledge and a conflict of interest.

Wartik highlights the objective difficulty involved in evaluating different technologies about which an evaluator possesses varying levels of insights and experience. He argues that unilaterally written comparative analyses should not be undertaken:

> "However, the fact that many of us aren't familiar enough with others' work to write comparative software engineering analyses is an important reason why we should reconsider the practice.[. . .]The simplest solution is perhaps the best – let's continue to reference related research, but let's skip subjective comparisons altogether." [War96]

Feitelson discusses the conflict of interest that evaluators will often find themselves in.

> "In the systems area, a common problem is the lack of objectivity. Inevitably, experimental and comparative studies are designed and executed by an interested party. They don't measure an independent, 'real' world, but rather a system they had created at substantial investment, and opposite it, some competing systems. In particular, there is practically no independent replication of the experiments of others. Thus reported experiments are susceptible to two problems: a bias in favor of your own system, and a tendency to compare against restricted, less optimized versions of the competition. Obviously, both of these might limit the validity of the comparison." [Fei06]

Zelkowitz and Wallace identify the same issue:

> "There are many examples where the developer of a technology wishes to show that it is effective and becomes both the experimenter and the

> subject of the study. [...] But all too often, the experiment is a weak example favoring the proposed technology over alternatives. As skeptical scientists, we would have to view these as potentially biased since the goal is not to understand the difference between two treatments, but to show that one particular treatment (the newly developed technology) is superior." [ZW97]

History teaches us how both problems can be fruitfully overcome for the mutual benefit of whole research communities. Some areas of computer science have been extraordinarily successful in setting standards for evaluation and creating common benchmarks. These are so widely accepted and used that, despite of some problems, they really define an independent largely uncontested reference against which all approaches in the field are commonly measured. Among these are the SPEC (Standard Performance Evaluation Corporation) benchmarks for computing performance[9] [GA95], the before mentioned series of TREC (Text REtrieval Conference) tasks in Information Retrieval[10] [Voo05] or the TPC (Transaction Processing Performance Council) benchmarks in the database community[11] [Ser93].

In each of these cases a consensus-based process, led by a number of key people, had been used to construct a benchmark that was endorsed by the whole research community. Consensus and community collaboration were the essential ingredients for the success of this process and the resulting benchmark. Henning summarizes by the example of SPEC how such a process leads to unbiased and fair benchmarks that are finally commonly accepted. He concludes:

> "It is nearly impossible to argue in the subcommittee, 'you should vote for 999.favorite because it helps my company.' Blatant efforts along these lines would backfire; subtle attempts may raise concerns about transparency.
>
> Of course, SPEC members who are vendor employees keep their employer's interests in mind. For example, an employee of a company that makes big-endian Unix systems makes sure that the playing field is not tilted in favor of little-endian NT systems. Arguments to level the playing field are always welcome and quickly attract support. But attempts to tilt the playing field just don't work." [Hen00]

Beyond fairness and acceptance, a community based consensus approach to benchmarking has another important advantage. SWS research is a young, still emergent field of science. In some ways, it is a visionary field of research. SWS are expected

[9] `http://www.spec.org/`
[10] `http://trec.nist.gov/`
[11] `http://www.tpc.org/`

to have a significant impact, but proposed technologies are not yet widely used in practice [HKZ08]. Thus, the problems associated with SWS are not yet well understood.

Brodie discusses this lack of a clear understanding of the problem space and compares SWS research with database research in the 1980's and the emergence of the TPC benchmarks:

> "It is easy to envisage services interacting dynamically to discover other services with which to negotiate, adapt, and compose, and then to invoke to achieve a requirement. It is quite another matter to specify correctness in this context, let alone achieve it in implementations.
>
> Almost three decades ago, the Next Generation of Computing, at the time, faced similar challenges. In the early 1980's the projected scale of relational databases was unimaginable, and like the Web of documents far exceeded its projections. As with our current Web of services we are facing unimaginable scale and complexity with novel, unproven technology and with few benchmarks. Now, as then, we require efficient, scalable solutions to problems for which we lack definitions of correctness. [...] Ideally semantic Web services benchmarks will contribute to the development and acceptance of semantic technologies just as relational benchmarks did for relational technology." [Bro08]

TCP, the primary benchmark for the database community, originated from a paper by so many contributors that "Anon et al." was given as author [ea85]. Incorporating the knowledge and experience of a whole community in a benchmark ensures that different views about important problems are considered and that rules for how to reliably and efficiently evaluate the proposed technologies are developed in a fair and unbiased way. Furthermore, community consensus processes have a built-in quality control, an invaluable advantage in a problem space without clear definitions of correctness. We will further discuss this aspect in the validation part of this thesis (Chapter 8).

1.2.3. Implicit Benefits of Community Evaluation Initiatives

As illustrated above, relying on a community approach to benchmark definition solves a number of practical problems. Above all, it avoids biases, promotes the acceptance of benchmark results and helps to gather the necessary technical knowledge for proper benchmark definition. However, further benefits have frequently been observed. Feitelson observes:

> "Progress is built from a combination of breakthroughs and small steps. The breakthroughs typically result from new insights that are based on

> cumulative experience. The small steps result from a choice between multiple candidates, just like evolution depends on the selection of the fittest among several variants.[...] Remarkably, this process can be accelerated artificially, by tapping on the competitiveness of humans in general and scientists in particular. This is done by setting up a common challenge, or competition. By getting multiple research groups to work on the same problem, and subjecting them to a common evaluation framework, it becomes easier to select the approach that promises the most rapid progress. This can then be used as the basis for the next round." [Fei06]

Sim and colleagues have extensively worked on this idea and developed a theory of benchmarking [Sim03, SEH03]. Their basic claim is that benchmarks operationalize scientific paradigms and thus advance the maturity of a scientific community. Furthermore, they advocate proactively pursuing community benchmarking in order to enjoy the positive side-effects associated with it:

> "Within a scientific discipline, the current paradigm captures the community consensus on which problems are worthy of study, and determines what are scientifically acceptable solutions. In this manner, paradigms convey implicit rules for working within the community along with the explicit rules or factual knowledge. [...]
>
> A benchmark operationalizes a paradigm; it takes an abstract concept and makes it concrete, so it can serve as a guide for action. The motivating comparison and task sample [of a benchmark] are a statement of the problems that are worth solving. The performance measures show which solutions are held in higher esteem. The benchmark also contains implicit information about how the problem ought to be solved. Like paradigms, benchmarks emerge though a process of scientific discovery and consensus. [...]
>
> The presence of a benchmark states that the community believes that contributions ought to be evaluated against clearly defined standards. The benchmark itself promotes the conduct of research that is collaborative, open and public.
>
> Collaboration in benchmarking occurs in two ways. During development, researchers work together to build consensus on what should be in the benchmark. During deployment, the results from different technologies are compared, which requires researchers to look at each other's contributions. Consequently, researchers become more aware of one another's work and ties between researchers with similar interests are strengthened.

> Evaluations carried out using benchmarks are, by their nature, open and public. The materials are available for general use, and often so is the technology being tested. It is difficult to hide the flaws of a tool or technique, or to aggrandize its strengths when there is transparency in the test procedures. Moreover, anyone could use the benchmark with the same tools or techniques, and attempt to replicate the results.
>
> These factors together, collaboration, openness, and publicness, result in frank, detailed, and technical communication among researchers. This kind of public evaluation contrasts sharply with the descriptions of tools and techniques that are usually found in conference or journal publications. A well-written paper is expected to show that the work is a novel and worthy contribution to the field, rather than share advice about how to tackle similar practical problems. Benchmarks are one of the few ways that the dirty details of research, such as debugging techniques, design decisions, and mistakes, are forced out into the open and shared between laboratories. [...]
>
> Given that benchmarks are indicative of the cohesiveness of a discipline on a technical and sociological level, we hypothesize that benchmarking can be applied proactively to advance the maturity of a scientific community, rather than simply enjoying this maturity as a side-effect. This hypothesis suggests that benchmarking can help whenever a research area needs to become more scientific, needs to codify technical knowledge, or needs to become more cohesive." [SEH03]

Sim provides extensive evidence and validation for this hypothesis through two case studies and further analysis of historic examples. From our experience in participating in and organizing community evaluation initiatives in the area of SWS, we strongly believe that the claims regarding the social benefits of benchmarking are particularly valid. During the design of benchmarks, researchers are forced to make their goals explicit to clearly state the standards that they oblige to be measured by. The execution of benchmarks fosters discussion on the level of technical quality of contributions and increases the chance of innovative but not yet established ideas to be picked up. It can move communication from a competitive marketing level to a level of technical collaboration. The application of different technologies to a common practical problem set enables a superior level of understanding for the characteristics of each others approaches which could not be reached based on the study of publications alone.

This thesis aims at laying a solid foundation for community-based comparative evaluations of SWS technologies. The corresponding thesis objectives will be detailed in the following section.

1.3. Thesis Objectives

The previous sections motivated the need for experimentation in computer science and the benefits of community-based comparative evaluations. This thesis will show that

- overall too little effort is being put into the experimental evaluation of SWS approaches,
- the few existing evaluation efforts lack a theoretic underpinning and critical appraisal of the chosen methodologies,
- a thorough and comprehensive discussion of the nature of evaluation in the area has been missing,
- the different dimensions to evaluate have never been comprehensively identified and in particular
- there are no standard benchmarks and evaluation methodologies for the comparative evaluation and assessment of SWS technologies.

Overall the important questions *what to evaluate*, *which criteria to use*, *how to measure those criteria* and *how to achieve reusability, comparability and impartiality* have not been answered in a well-founded way. This motivates the objectives of this thesis:

Objective 1: Development of a comprehensive and well-founded conceptual model for SWS technology evaluation.

- Identify evaluation dimensions, i.e., the criteria to evaluate.
- Identify evaluation requirements to promote and ensure evaluation quality.

Objective 2: Provide reference benchmarks for selected evaluation criteria and use cases to solve concrete benchmarking needs in the area.

- Identify measures for selected criteria.
- Design and implement measuring instruments to assess a system with respect to these measures.
- Develop and establish methodologies to obtain measurements and conduct an evaluation.

SWS evaluations need to allow the meaningful comparison of different approaches relying on different formalisms to allow establishing a common understanding of the pros and cons of the various technologies. As motivated above, the participation and agreement of the community during the development of evaluation procedures is essential for ensuring evaluation usefulness, reliability, fairness and acceptance. This leads to the major requirements for the developed solution and the methodology to follow on the path to this solution.

Main Requirements: Impartiality, community participation, continuous application.

- Ensure the applicability of the evaluation framework and the developed benchmarks to different technologies and avoid unnecessary prerequisites and any biases to particular approaches.
- Promote a culture of collaboration for enabling community input and feedback during the process of benchmark development.
- Establish structures to foster the dissemination of the benchmark and the continuous co-evolution of the benchmarking efforts and the scientific community.

In the following section we will detail the contributions of the thesis and the approach for achieving the thesis objectives while meeting the stated requirements. The chapter will be concluded by an overview of the thesis structure in Section 1.5.

1.4. Research Contributions and Solution Approach

The main contributions of this thesis are the following. A conceptual model for SWS technology evaluation is provided. This model defines possible criteria dimensions and relates them to each other. It allows classifying evaluation approaches and putting them into context. This model is derived by applying the Goal-Question-Metric approach, a methodology from software engineering for operationalizing design goals into quantifiable metrics.

Furthermore, the conceptual model comprises a detailed catalogue for requirements to SWS technology evaluation. This requirements catalogue promotes evaluation quality and supports a meta-evaluation of SWS technology evaluations. It has been derived from evaluation standards published by the German Society for Evaluation and adapted and concretized with input from a literature review on evaluation requirements in related areas.

Having introduced the conceptual model for SWS technology evaluation, a structured analysis of the state of the art has been performed. This analysis discusses the evaluation criteria covered by existing approaches and comprises a meta evaluation of their current shortcomings. Defining, implementing and executing benchmarks for all criteria dimensions and all context environments identified by the conceptual framework is far beyond the scope of this thesis. Thus, based upon the analysis of previous work in the area, choices about the concrete benchmarking contributions within the scope of this thesis had to be made. These will be motivated in more depth in Section 4.5, but already introduced here.

The provided benchmarking contributions are threefold:

- a solution for obtaining meaningful test data for SWS technology evaluation,
- a benchmark for assessing the functional scope of SWS frameworks and
- a benchmark for evaluating the retrieval correctness of SWS matchmaking approaches.

Other problems are partially covered, but the mentioned three form the main concrete benchmarking contribution of this thesis. A solution for how to obtain meaningful test data is considered to be fundamental to almost all SWS benchmarking problems and was therefore selected as the first problem to be tackled. The two benchmarking choices were motivated by the emergence of two community evaluation initiatives that started focusing on the corresponding problems. I.e., they reflect what the community chose to be the most interesting, important and feasible benchmarking problems. Furthermore, involvement in the organization of these initiatives enabled tighter and more effective interaction with the wider community. As argued in the previous section, this is viewed as the most crucial prerequisite for successful benchmarking.

Both benchmarks were approached through an iterative process. Existing work was analyzed, shortcomings identified and improvements devised. The setup of the benchmarks was repeatedly discussed in the wider community and the benchmarks were then adapted based on the feedback from the community. This process ensured the benchmark's relevance to the community's benchmarking needs, unbiased treatment of all potentially relevant technologies and generally a proper quality assurance in an area where straightforward quality standards are not available.

Both benchmarks were executed (the first one multiple times) as part of open community evaluation campaigns. Although participation in either benchmark required significant effort from the participants, they attracted wide participation. This once more illustrates the need for such benchmarks, but also the appreciation of the benchmarks' quality.

Finally, the contributions of the thesis are validated. The conceptual model for SWS technology evaluation is discussed with respect to its usefulness and completeness for structuring evaluation approaches, relating them to each other and assessing their quality on an abstract level. An evaluation of the three concrete benchmarking contributions is more difficult. Ideally, a meta-evaluation would have to show that the provided artifacts served the evaluation purposes, i.e. enabled more efficient test data definition and provided reliable and cost-effective assessments of different technologies, thus effectively helping to advance them. This is almost impossible within the time frame of a thesis.

Validation of the concrete benchmarking contributions is thus approached by discussing the concrete advancements of the provided benchmarks over the previous state of the art and by examining the role of the quality assurance provided by the wider community.

1.5. Thesis Structure

The thesis is structured according to the approach outlined above. To provide an overview, the structure of the thesis in terms of parts and chapters is depicted below and also shows, where applicable, references to publications covering contributions of the respective chapters.

I Foundation

1. Introduction
2. Background
3. State of the Art

II Evaluation of Semantic Web Service Technology

4. Conceptual Model for SWS Technology Evaluation [KKRK10, KKRPK08, KLKR07]
5. Test Data for SWS Evaluation [KLKR07, KKR08b, KKR08d, KKRK08a, KKRK08b]
6. Benchmarking the Functional Scope of SWS Discovery Frameworks [PKMS08, PLZM08, KKRK06a, KKRK06b, KKR06a, KKR07c, KKR07a, KKR07b]
7. Benchmarking SWS Matchmaking [KKR08a, KKR09, KKR10]

III Finale

8. Validation [PKMS08, PLZM08]

9. Conclusions and Outlook

   References

Appendix

A. Analysis of SWS Evaluation Campaigns by Evaluation Requirements

B. Additional Information on the Functional Scope Benchmark

C. Additional Information on the SWS Matchmaking Benchmark

Subsequent to the introduction, the background in the fields of Semantic Web, Web Services, Semantic Web Services and Evaluation and Benchmarking is explained in Chapter 2. Based upon this background, Chapter 3 provides an overview of the state of the art directly related to this thesis.

The next four chapters contain the core contributions of the thesis: the conceptual model for SWS technology evaluation (Chapter 4), the work on test data for SWS evaluation (Chapter 5), the benchmark for the functional scope of SWS discovery frameworks (Chapter 6) and the benchmark for SWS matchmaker evaluation (Chapter 7). The work is validated in Chapter 8 and Chapter 9 concludes the thesis. Finally, the appendix makes some additional extended information available for reference.

CHAPTER 2

Background

> When I took office, only high energy physicists had ever heard of what is called the Worldwide Web... now even my cat has its own page.
>
> *(Bill Clinton)*

In this chapter the basic background for this thesis will be briefly introduced. The thesis is concerned with Evaluation and Benchmarking (Section 2.4) of Semantic Web Service technologies (Section 2.3) which result from applying technologies from the Semantic Web (Section 2.1) to Web Services (Section 2.2).

We will motivate below that we follow an approach to evaluation as fitness for purpose. Thus, this thesis pursues a functional approach to evaluation. Technologies are not evaluated with respect to some intrinsic technical properties, but with respect to their suitability to solve concrete problems. An in-depth understanding of the internal details of specific SWS approaches is thus helpful, but not essential. Therefore, the coverage of the technical background of semantic web services will be kept rather short. The background presented in this chapter will be complemented by a detailed report on the state of the art in SWS evaluation in Chapter 3.

2.1. The Semantic Web

The World Wide Web has grown to become the "the universe of network-accessible information, the embodiment of human knowledge"[1], yet, as of today, much of the information available can only be comprehended and used in a meaningful way by

[1] `http://www.w3.org/WWW/`

humans. According to Tim Berners-Lee, the father of the web, this is in contrast to the Web's original vision:

> "The Web was designed as an information space, with the goal that it should be useful not only for human-human communication, but also that machines would be able to participate and help. One of the major obstacles to this has been the fact that most information on the Web is designed for human consumption, and even if it was derived from a database with well defined meanings (in at least some terms) for its columns, that the structure of the data is not evident to a robot browsing the web." [BL98]

Frank van Harmelen discusses several problems that result from this issue. Among these are low precision or low recall in search for information, sensitivity to vocabulary arising from imprecision of language or the inability to integrate information from different sources [AvH04]. Peter Mika analyzes the reason for resulting limitations of current web usage:

> "... we deal with a knowledge gap: what the computer understands and able to work with is much more limited than the knowledge of the user. The handicap of the computer is mostly due to technological difficulties in getting our computer to understand natural language or to 'see' the content of images and other multimedia. Even if the information is there, and is blatantly obvious to a human reader, the computer may not be able to see anything else of it other than a string of characters. In that case it can still compare to the keywords provided by the user but without any understanding of what those keywords would mean." [Mik07]

In order to overcome this knowledge gap and unleash the full potential of the web, the Semantic Web was proposed as an extension of the current one, in which information is given well-defined meaning, better enabling computers and people to work in cooperation [BLHL01]. The Semantic Web is realized by applying advanced knowledge technologies to the Web and distributed systems in general [Mik07]. Knowledge representation languages and techniques are used to formalize a domain of discourse in order to make necessary background knowledge accessible to programs. Logic is used to establish correctness of such domain models and to infer implicitly stated knowledge by means of inference rules. Annotations and meta data are used to disambiguate information by linking it to unambiguous concepts provided in some shared background knowledge. Figure 2.1 shows a more detailed overview of the current corresponding technology stack underlying the Semantic Web. Its components will be briefly introduced in the following.

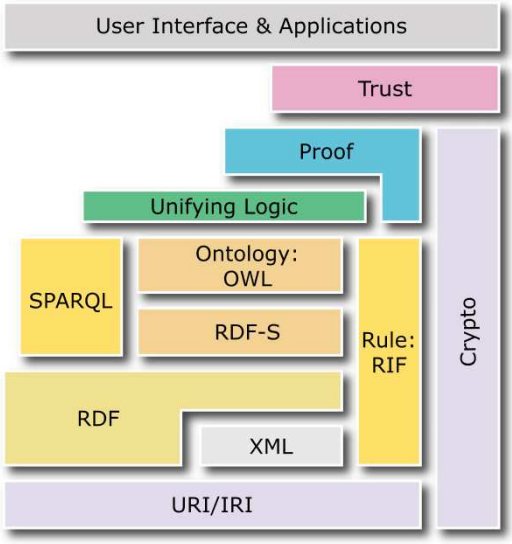

Figure 2.1.: Technology stack of the Semantic Web.
(Source: `http://www.w3.org/2007/Talks/0130-sb-W3CTechSemWeb/#(24)`, accessed December 2009.)

URI/IRI Uniform Resource Identifier (URI)[2] and Internationalized Resource Identifier (IRI)[3] provide unique and unambiguous names for things.

XML eXtensible Markup Language (XML)[4] provides syntactic means to add arbitrary structure to documents without specifying what the structures mean.

RDF The Resource Description Framework (RDF)[5] provides such meaning by relating concepts to triples much like the subject, verb and object of an elementary sentence. RDF triples assert properties of things and relations among objects, thus forming networks of concepts.

RDF-S RDF Schema (RDF-S)[6] allows limiting the usage of RDF by indicating which concepts may be related in which way to other concepts. In particular it supports the definition of classes, hierarchies of classes and restrictions on property domains (allowed subjects) and ranges (allowed values).

[2] `http://tools.ietf.org/html/rfc3305`
[3] `http://www.ietf.org/rfc/rfc3987`
[4] `http://www.w3.org/XML/`
[5] `http://www.w3.org/RDF/`
[6] `http://www.w3.org/TR/rdf-schema/`

OWL The Web Ontology Language (OWL)[7] adds another semantic layer on top of RDF-S. It allows modeling a domain of interest by representing classes and instances of entities in this domain, their attributes and the relations between the classes, as well as constraints on the sets of valid instances by means of description logics. Such formal, explicit specification of a shared conceptualization is called ontology.

SPARQL The SPARQL Protocol And RDF Query Language (SPARQL)[8] provides means for querying RDF data much like the Structured Query Language (SQL) allows querying relational data.

RIF The Rule Interchange Format (RIF)[9] provides means to interchange logical knowledge inference rules.

Unifying Logic provides a logic layer that unifies the underlying logics for ontologies, rules, queries, and RDF-S. Reasoning is used to establish correctness and infer implicit knowledge from explicit facts.

Proof explains and justifies the process of logical inference.

Crypto provides means to ascertain the identity of an agent or a source of information or to protect information against unauthorized access.

Trust establishes authentication and trustworthiness of information and agents.

User Interfaces and Applications enable programs and humans to leverage the underlying layers.

Further comprehensive information about the Semantic Web and its underlying technologies is available in the standard literature [AvH04, FHL05, HRK09].

2.2. Service Orientation

Service orientation is a new computing paradigm that gained popularity within the last decade. We will first introduce the general idea of service oriented architectures before Web Services, the most prominent implementation technology of the paradigm will be covered..

[7] http://www.w3.org/2004/OWL/
[8] http://www.w3.org/2009/sparql/wiki/Main_Page
[9] http://www.w3.org/2005/rules/

2.2.1. Service Oriented Architectures

Service Oriented Architectures (SOAs) refer to an architectural style for designing distributed information systems. A SOA is essentially a collection of modular, self-contained, self-describing, interoperable functions, called services:

> "Web services are a new breed of web application. They are self-contained, self-describing, modular applications that can be published, located, and invoked across the Web. Web services perform functions that can be anything from simple requests to complicated business processes. A sample web service might provide stock quotes or process credit card transactions. Once a web service is deployed, other applications (and other web services) can discover and invoke the deployed service." [Tid00]

As is indicated by the quote above, services are based on a number of long established principles from software engineering, among them:

Encapsulation: Services provide access to a modular, coherent and self-contained piece of functionality that does not depend on the state of other services.

Network accessibility: Services can be accessed remotely over a network using open standards.

Platform independence: Services communicate using open protocols and platform-independent data representation formats.

Service Contract: Services adhere to a communication contract that is defined through their interface.

Loose Coupling: Services minimize dependencies beyond the contract established by their interface and thus provide interoperability between applications running on a variety of platforms and frameworks.

By stressing the principles above, SOAs promise to improve maintainability of information systems, facilitate smoother integration and interoperability of distributed systems and generally make IT more agile [Erl05, MSJL06].

Some terms are commonly used in the context of SOAs. The process of locating a service that offers a desired functionality is called *service discovery*. The task of comparing a service offer description with a service request description to determine whether or to which degree the offer is capable of providing the requested functionality is called *service matchmaking*. Replacing an abstract service interface stub by a concrete service implementation is referred to as *service binding*. The task of coordinating various services to a well defined process is named *service composition*.

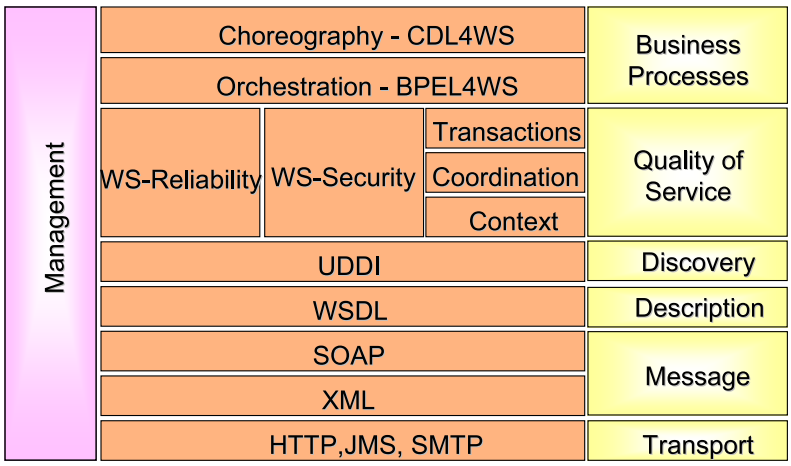

Figure 2.2.: Web service technology stack (from [Pap08])

The global view on the services being coordinated and the logic that coordinates them is called *service orchestration*. In contrast, the communication pattern and sequence of messages between services is called *service choreography*. Finally, the resolution of data or process mismatches in the interfaces of services being composed is called *data* and *process mediation*.

2.2.2. Web Services

The term *service* in the sense defined above and the term *web service* are sometimes used interchangeably. However, more often, the term *web service* refers to a particular implementation of a service, characterized through a stack of concrete so called web service standards, the most important of which are depicted in Figure 2.2.

Within the context of this thesis, XML, SOAP and WSDL are of greater importance than the other standards.

XML/SOAP The Simple Object Access Protocol[10] defines the schema and encoding of messages on top of an XML syntax. Web services are primarily characterized by the SOAP messages they are able to consume and send.

WSDL The Web Service Definition Language 2.0[11] defines an XML based schema for describing web services in terms of the messages it sends and receives. *Messages* are defined using the XML Schema type system. An *operation* associates a message exchange pattern with one or more messages. An *interface*

[10] `http://www.w3.org/TR/soap/`
[11] `http://www.w3.org/TR/wsdl20/`

defines a group of related operations. These elements define the service on an abstract level without any commitment to transport or wire format. At a concrete level, a *binding* specifies transport and wire format details for one or more interfaces and an *endpoint* associates a network address with a binding. Finally, a *service* groups together endpoints that implement a common interface.

Even though other technical implementations like REST (REpresentational State Transfer) [Fie00] based services have recently gained some popularity, web services based on the technology stack depicted in Figure 2.2 are the dominant way of implementing services on the web. They have proven to be an efficient way of enabling reliable integration of distributed systems across organizational boundaries and are highly used especially in the area of enterprise application integration [Pap08].

However, WSDL descriptions exclusively define syntactic aspects of services; they do not comprise the semantics of a service, i.e., what a service does. Such semantics are only implicitly available through the naming of elements (messages, types, interfaces, etc.) or via natural language text annotations of the WSDL elements. This information can not be processed automatically in a reliable way which renders the automation of service discovery, mediation, composition or binding effectively infeasible. In short, despite of their success, web services suffer from the same problem as the general Web: their full potential can not be leveraged as long as human interaction is necessarily involved to reliably interact with them.

2.3. Semantic Web Services

In order to facilitate the partial or full automation of discovery, composition, binding, mediation and execution of web services, an application of the principles of the semantic web to web services has been proposed [Hen01, MSZ01]. The relationship between the web, the semantic web, web services and semantic web services is depicted in Figure 2.3. Web services transform the web from being static to being dynamic. Semantics transform the web from being syntactic to having well defined meaning for machine processing. Finally, semantic web services aim at creating a dynamic, machine processable information space that agents can autonomously browse and interact with [SAG07].

SWS rely on two building blocks, the formal descriptions that make the semantics of the services accessible to computers and the algorithms that leverage those semantics to facilitate the automation of the service computing tasks listed above. Below, we will give a brief introduction to semantic service descriptions and semantic service processing.

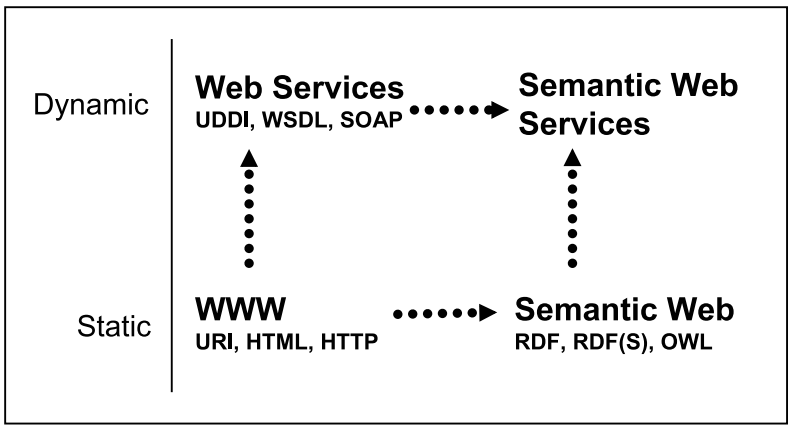

Figure 2.3.: The semantic web service vision: Bringing the Web to its full potential (from `http://www.wsmo.org/TR/d17/v0.2/`)

2.3.1. Semantic Service Descriptions

A variety of approaches to semantic service descriptions has been proposed [SAG07, Klu08b]. These vary in the formalism employed for the semantic annotations, the way the formalism is used to model services and the elements of the services covered by the descriptions. Most structured semantic description approaches can be expressed in terms of inputs, outputs, preconditions and effects (IOPE). The concept behind IOPE is that a service can be invoked if its preconditions, i.e., assumptions about the state of the world and the information space prior to the service invocation hold. The invocation of the service then transforms the world and the information space based upon the provided inputs to a state where the assumptions modeled in the postconditions hold and furthermore delivers the specified outputs.

Please note that structured IOPE are not the only way of modeling semantics for services. More simplistic approaches may formalize services monolithically [Klu08b] or also in a very lightweight fashion, for instance via a set of semantically disambiguated business classifiers or other keywords. Furthermore, besides IOPE, non-functional information like trust, reputation, quality of service etc. and provenance information such as the service name, its business domain and provider may be additionally or alternatively modeled.

The goal of this thesis is to enable comparative evaluations of a range of technologies as wide as possible. One of the core evaluation questions concerns the appropriateness of different description approaches to different problems. Thus the thesis tries to avoid assumptions about the technologies under evaluation as much as possible. Instead of investigating the intrinsic properties of different description

approaches, for instance, the expressivity of the employed logic, the thesis pursues an experimental approach to an evaluation as fitness for purpose. Rather than discussing technologies in theory, they are evaluated with respect to the results that are achieved when applying them to concrete problems. We will therefore only briefly introduce the three main approaches to SWS descriptions, namely SAWSDL, WSMO and OWL-S. A more comprehensive treatment of SWS description approaches is available in the literature [Klu08b, SAG07, KTSW08].

SAWSDL

Semantic Annotations for WSDL and XML Schema (SAWSDL)[12] is a W3C Recommendation since 2007. It provides a simplistic standard for attaching semantic annotations to WSDL documents by means of *model references* and *schema mappings*. The *modelReference* attribute defined by SAWSDL may be attached to any WSDL element (interface, operation, fault, complex type, ...) and links the element it is attached to with concepts in a semantic model via a set of URIs. The *liftingSchemaMapping* and *loweringSchemaMapping* attributes define hooks to attach transformation rules between the syntactic XML consumed and produced by the described service and the semantic model used to describe the service. The process of converting XML data to the semantic model is called *lifting* while the opposite transformation is called *lowering*.

SAWSDL builds directly on the well established WSDL standard and thus integrates smoothly with well established WS technologies. However, it is simplistic in that it does not standardize the type of semantic model used in the model references. SAWSDL allows referring to arbitrary domain models in arbitrary formalisms and no clear semantic is defined if multiple model references are used for the same element. This makes the meaningful processing of SAWSDL descriptions particularly challenging unless conventions about the admissible use of semantics are agreed upon on top of the SAWSDL standard. Furthermore, while WSDL provides natural hooks for describing the inputs and outputs of a service (IO), it does not specific hooks for preconditions or effects (PE). Corresponding annotations can be attached to operations, but again, conventions on how to differentiate between preconditions and effects and how these are to described precisely have to be agreed upon on top of the SAWSDL standard.

OWL-S

OWL-S defines an upper ontology for services[13]. As is apparent from its name, OWL-S uses OWL as description formalism. OWL-S distinguishes among the *service*

[12] `http://www.w3.org/TR/sawsdl/`
[13] `http://www.w3.org/Submission/OWL-S/`

profile, the *service model* and the *service grounding*. The service profile describes what a service does, the service model how the service works internally and the service grounding how it can be accessed.

The profile defines attributes for provenance information, service inputs and outputs as well as preconditions and conditional effects (called results). Inputs, outputs, preconditions and effects are references to the service process, a subconcept of the service model.

Service processes may be used to define the internal workings of a service by means of workflow control flow statements. Furthermore the process defines the previously mentioned IOPEs. While IOs are defined as concepts from an OWL ontology, PEs refer to arbitrary expressions specified in some logic formalism. However, OWL-S does not presuppose the formalism being used, from the viewpoint of OWL-S such expressions are merely literals.

Finally, the service grounding links the semantic description of the service from the profile and the process to the service's implementation. The approach is to define a specialized grounding class for each type of service implementation, like the *WsdlGrounding* for WSDL style web services. Those grounding classes define the attributes to link the IOs with the proper XML messages and the process model with the operations to be invoked in proper sequence.

WSMO

WSMO[14] defines a framework around semantic web services. It consists of three subprojects. The *Web Service Modeling Ontology (WSMO)* is an ontology for describing semantic web services. The *Web Service Modeling Language (WSML)* is a family of languages based on Description Logic and Logic Programming with varying levels of logical expressiveness and complexity. Finally, the *Web Service Execution Environment (WSMX)* is a reference implementation of WSMO using the WSML family of languages. WSMX uses WSMO as its conceptual model and defines its own execution semantics, architecture and implementation.

WSMO is based on four conceptual elements. *Ontologies* provide the formalization of the information used by all other components. *Web services* represent a service's functional and behavioral aspects which need to be semantically described in order to allow semi-automated usage. Unlike OWL-S or SAWSDL, WSMO makes an explicit distinction between web services and *goals*. The latter specify the objectives that a client has when invoking a web service. Finally, WSMO adopts a mediator-centric approach to semantic services. *Mediators* describe elements that help overcoming structural, semantic or conceptual mismatches that appear between the different components that build up a WSMO description. Mediators are used

[14] `http://www.wsmo.org/`, `http://www.w3.org/Submission/WSMO/`

to mediate among ontologies (*OOMediators*), goals (*GGMediators*), between a goal and a web service (*WGMediators*) and between several web services that need to collaborate (*WWMediators*).

Web services are primarily defined by a capability, one or multiple interfaces and some additional nonfunctional properties. The capability defines the functionality of a service in terms of pre- and postconditions, assumptions and effects. These correspond to the commonly used PE, but explicitly distinguish between the state of the information space (pre- and postcondition) and the real world (assumption and effect). An interface of a web service provides further information on how the functionality of the web service is achieved via a description of the communication pattern that allows consuming the functionality of the web service (*choreography*) and the description of how the overall functionality of the web service is achieved by means of cooperation of different web service providers (*orchestration*). Goals are defined similarly to web services by describing the requested capability, the requested interface and additional nonfunctional properties.

2.3.2. Semantic Service Processing

The semantic descriptions described above allow the partial or full automation of the tasks necessary to consume services. This concerns all tasks listed in Section 2.2.1 but the tasks most commonly addressed with SWS are service matchmaking/discovery and service composition. These will be briefly introduced in the following. For a more thorough discussion of the topic and coverage of the variety of algorithms in the area the interested reader is referred to [Pee05, Klu08a, SAG07, KTSW08].

Semantic Service Matchmaking

Semantic service matchmaking refers to the task of comparing a service request description with a service offer description to determine whether or to which degree the offer is available to provide the functionality sought by the request. Typically, this term is used interchangeably with the term *service discovery*. With full IOPE descriptions, service matchmaking usually concerns comparing whether the inputs of the service are available and its preconditions are fulfilled as well as whether the effect of the service creates the desired world state and its outputs offer the desired information. Non-functional properties and preferences may be integrated in the above process and entirely different modes of comparison are possible depending on the chosen model for the service descriptions.

Klusch, for instance, provides a classification of SWS matchmakers that differentiates between the technique used for comparison (logic based, non-logic based and hybrid approaches) and the elements of descriptions that are leveraged for com-

parison. Here, he distinguishes among profile and process based matchmakers and further differentiates the profile based matchmakers into non-functional, functional and combined matchmakers. Finally, functional matchmakers may be based upon comparing inputs and outputs (IO), effects (E), preconditions and effects (PE) or full IOPE information [Klu08a].

This gives a good impression of the variety of approaches encountered in the area. In order to promote an unbiased comparison, this thesis will generally take a functional black box approach to this variety. The benchmarks contributed by this thesis will specify concrete tasks or problems to be solved, but avoid making assumptions about the necessary underlying model of descriptions or processing algorithms.

An additional clarification of the term service matchmaking is necessary. Service matchmaking is an imprecise term in that it may refer to two quite distinct use cases. This is due to the semantic ambiguity of the term *service* [Pre04]. On the one hand, it may refer to the process of provisioning value in some domain. On the other hand, it may refer to a software entity (e.g., a web service) able to perform this process. With this respect, a *web service* provides access to *services*. Using this terminology, a user may be interested in discovering web services, typically to embed them in some application, or in discovering services, typically to consume some value [FKL+05]. As an illustrating example, a user may be interested in locating a service that provides weather forecasts within the US (web service discovery), or in getting to know the current weather forecast for San Francisco (service discovery). Service discovery sometimes requires additional interaction with a web service (called *negotiation*) on top of web service discovery to determine whether a potentially suitable web service actually provides the desired service at the desired terms. This process may also comprise determining concrete input values, i.e., the configuration of the web service that is necessary to retrieve the desired service.

Semantic Service Composition

Semantic service composition refers to the task of assembling multiple web services and coordinating them such that the resulting process provides a desired functionality. Automated semantic service composition is usually achieved through means of AI planning techniques where services are treated as planning operators or actions. A comprehensive overview of the various techniques and approaches is available in [Pee05, Klu08a].

Semantic service composition and service discovery are closely related tasks. On the one hand, service composition requires service discovery if the component services to be coordinated are not known beforehand. On the other hand, service discovery may involve service composition if no single service is found that provides the desired functionality, but several services combined achieve the desired effects.

Service composition may be horizontal (the goal is decomposed into sub goals, each of which is achieved by a component service), vertical (the goal is achieved by a chain of subsequent service calls each building upon the results of the previous call) or complex (the goal is achieved by a complex workflow-like interaction among the component services).

2.4. Evaluation and Benchmarking

After having introduced the technologies that this thesis is concerned with, namely the Semantic Web, Service Orientation and Semantic Web Services, we now turn to clarifying the notions of evaluation and benchmarking of technologies as used throughout this thesis.

2.4.1. Evaluation in Computer Science

We start with stating our definition of evaluation in general, and benchmarking of technology in particular. With respect to evaluation, we adopt a definition from the "Evaluation Standards" by the German Evaluation Society (DeGEval[15]):

> "Evaluation is the systematic investigation of an evaluand's worth or merit. Evaluands include programmes, studies, products, schemes, services, organisations, policies, technologies and research projects. The results, conclusions and recommendations shall derive from comprehensible, empirical qualitative and/or quantitative data." [Bey03]

The "Guiding Principles" of the American Evaluation Association[16] provide an alternative definition that stresses the evaluation's purposes, which include:

> ... bettering products, personnel, programs, organizations, governments, consumers and the public interest; contributing to informed decision making and more enlightened change; precipitating needed change; empowering all stakeholders by collecting data from them and engaging them in the evaluation process; and experiencing the excitement of new insights." [SNSW94]

These notions of supporting improvement, preparing decision making and generally broadening of knowledge are also mentioned in the DeGEval standards. Both standards deal with evaluation in the broadest possible sense without a specific focus on certain fields. As motivated in Section 1.2, this thesis is concerned with

[15] `http://www.degeval.de`
[16] `http://www.eval.org`

the evaluation of new technology with the primary objective of fostering scientific progress. Feitelson compares the role of such experimental evaluation in engineering with the role of experimental tests in other sciences [Fei06].

In the traditional scientific method, an observation leads to a hypothesis or model. This model is used to make a concrete prediction about the nature of an observed phenomenon. The prediction is checked through experimental tests, leading to the verification, rejection or modification of the hypothesis or model.

In engineering, a new idea how to solve or improve upon a concrete problem leads to a new system design. The design is realized in a concrete implementation. An experimental evaluation is used to verify whether the new idea, incarnated by the implemented system, delivers the expected solution or improvement. Based upon the results of the experimental evaluation the system's design is verified, rejected or modified.

This highlights that in engineering, experimental evaluation serves the purpose of verifying an expected improvement. Naturally, such verification may also include quantifying an improvement with respect to alternative designs or exploring the nature and causes of the improvement to support further system enhancements.

This notion is elaborated by Gediga et al., who characterize the potential goals of software evaluation:

> "In the domain of software evaluation, the goal can be characterized by one or more of three simple questions:
>
> 1. 'Which one is better?' The evaluation aims to compare alternative software systems, e.g. to choose the best fitting software tool for given application. [...]
> 2. 'How good is it?' This goal aims at the determination of the degree of desired qualities of a finished system. [...]
> 3. 'Why is it bad?' The evaluation aims to determine the weaknesses of a software such that the result generates suggestions for further development." [GHD02]

Gediga et al. continue introducing the notions of *formative* (first two goals) versus *summative* (third goal) evaluation:

> "...summative evaluation [...] does not offer constructive information for changing the design of the system in a direct manner. [...] In contrast, the goals of formative evaluation are the improvement of software and design supporting aspects. It is considered the main part of software evaluation, and plays an important role in iterative system development." [GHD02]

From the viewpoint of scientific evaluations, both types of evaluation are of interest. We want to become able to determine the best technique to solve particular problems or subproblems ("Which one is better?"), we want to know whether and how much new technologies improve over the state of the art ("How good is it?") and finally and most importantly, we want to detect weaknesses and their causes to determine the problems to devote further research to ("Why is it bad?"). The term *evaluation* will refer to these objectives throughout this thesis.

2.4.2. Benchmarking as a Method of Experimental Evaluation

We now turn to benchmarking as a specific method of experimental evaluation, relate it to other methods and highlight the characteristics that are specific and central to benchmarking as opposed to other forms of evaluation.

The Merriam-Webster defines the term *benchmark* as "a standardized problem or test that serves as a basis for evaluation or comparison"[17].

Kitchenham has developed a classification of evaluations methods that distinguishes three quantitative methods (quantitative experiment, quantitative case study and quantitative survey), five qualitative methods (qualitative screening, qualitative experiment, qualitative case study, qualitative survey and qualitative effect analysis) and benchmarking. She highlights the use of *standard tests* for the *direct comparison* of alternatives as the distinguishing characteristics of benchmarking:

> "Benchmarking is a process of running a number of standard tests/trials using a number of alternative tools/methods (usually tools) and assessing the relative performance of the tools in those tests" [Kit96].

As is evident from these two quotes, standardized direct comparison of alternatives is the primary distinguishing characteristic that differentiates benchmarking from other types of evaluation.

Zelkowitz and Wallace have also developed a categorization of software engineering validation techniques for evaluating processes as well as products. They distinguish among observational, historical and controlled methods. The controlled methods are differentiated into replicated experiment, synthetic environment experiment, dynamic analysis and simulation. Zelkowitz and Wallace characterize benchmarking as a specific implementation of a dynamic analysis:

> "We can also look at controlled methods that execute the product itself. We call these dynamic analysis methods. Many instrument the given product [...] in such a way that features of the product can be demonstrated and evaluated when the product is executed. [...]

[17] `http://www.merriam-webster.com/dictionary/benchmark`

> The major advantage of this method is that scripts can be used to compare different products with similar functionality. The dynamic behavior of product can be determined often without a need to understand the design of the product itself. Benchmarking suites are examples of dynamic analysis techniques. These are used to collect representative execution behavior across a broad set of similar products." [ZW98b]

This stresses that benchmarking can be used as a black box approach without necessarily dealing in depth with the internal workings of the system under evaluation. As such it is typically a *functional evaluation of a fitness for purpose*, as is highlighted by Sim:

> "... a benchmark is defined as a standardized test or set of tests used for comparing alternatives. A benchmark has three components, a Motivating Comparison, a Task Sample, and Performance Measures." [Sim03]

The Motivating Comparison captures both, the actual comparison being made and the motivation for making this particular comparison, i.e., for performing the benchmarking. The task sample refers to the selection of tasks that serve as the basis for tool comparison. Finally, the performance measures assess the relative worth or merit of a tool. Sim remarks:

> "Performance is not an innate characteristic of the technology, but is the relationship between the technology and how it is used. As such, performance is a measure of fitness for purpose." [Sim03]

Finally, Wohlin and colleagues add another important aspect by remarking:

> "It is important to note that benchmarking is a continuous improvement process rather than a competitive comparison." [WAP+02]

Sim further elaborates on this notion of continuity and replication:

> "A convenient feature of benchmarking is that replication is built into the method. Since the materials are designed to be used by in different laboratories, people can perform the evaluation on various tools and techniques, repeatedly, if desired. Also, some benchmarks can be automated, so a computer does the work of executing the tests, gathering the data, and producing the Performance Measures." [Sim03]

By now, we have assembled the central characteristics of benchmarking as the term is used throughout this thesis:

A benchmark is a *set of well-defined problems* from a certain domain consisting of *sample data and sample tasks* to perform on that data. Benchmarks support the *repeated, standardized comparison of alternatives*. Benchmarking is the process of performing such comparison of alternative systems or tools based on their *fitness for purpose* of solving the given benchmark problems. It is a *continuous process* with the primary objective of *gathering knowledge* about and supporting the *iterative improvement* of the benchmarked tools.

This definition concludes the coverage of the technical context and background of this thesis.

CHAPTER 3

State of the Art

There's a way to do it better
— find it.

(Thomas Edison)

The previous chapter introduced the necessary technical background of this thesis. This chapter complements that introduction by a coverage of the state of the art in SWS technology evaluation and closely related work. Section 3.1 reports about SWS technology evaluation. Section 3.2 briefly reports about related work from the areas of software engineering, ontology evaluation, ontology alignment, reasoning and triple stores as well as AI (Artificial Intelligence) planning. Finally, Section 3.3 concludes the chapter.

3.1. Semantic Web Service Evaluation

In the following, we will introduce the three community evaluation initiatives in the area of SWS evaluation (Sections 3.1.1 – 3.1.3). Subsequently, evaluations performed as part of SWS project validations (Section 3.1.4) as well as other SWS evaluation efforts (Section 3.1.5) will be covered.

3.1.1. Semantic Web Services Challenge

The Semantic Web Service Challenge [PLZM08, PKMS08] is an initiative aiming to "develop a common understanding of various technologies intended to facilitate the automation of mediation, choreography and discovery for Web Services using semantic annotations. The intent of this challenge is to explore the trade-offs among

existing approaches. Additionally we would like to figure out which parts of problem space may not yet be covered"[1].

The SWS Challenge was founded by STI Innsbruck (Austria, previously called DERI Innsbruck) and Stanford Logic Group (CA, USA) and launched in March 2006 as an activity of the Knowledge Web EU project[2] (which ran till the end of 2007) [VLZP06]. It is meanwhile jointly organized by a group of people from Open University (UK), STI Innsbruck (Austria), the candidate's research group at University of Jena (Germany), University of Potsdam (Germany), Technical University of Dortmund (Germany) and Stanford University (CA, USA). Since 2007, the candidate has coorganized the SWS Challenge and part of this thesis work (Chapter 6) was performed under the umbrella of the SWS Challenge. Therefore, the SWS Challenge will be described and discussed in detail in Chapter 6 and only a relatively brief description will be given here.

The SWS Challenge provides a certification service for semantic SOA technologies. Furthermore, it attempts testing the assumption that the use of formal semantics increases programmer productivity by making the resulting programs more flexible and adaptable to change. With this respect it also aims to verify whether semantic technologies constitute an improvement over traditional software engineering methods. The approach is to provide a set of common scenarios in a publicly available testbed. Prospective participants are challenged to show what their Web service mediation, discovery, and composition technologies can really do by solving the provided problem scenarios.

Scenario descriptions are provided in natural language English text. This is motivated by the insight that the usage of a particular formalism for describing services and scenarios already implies the solution to a large degree. In order to provide a level evaluation ground and avoid presupposing a particular solution approach, the SWS Challenge organizers thus decided not to provide formal descriptions but only natural language ones.

So far, two mediation scenarios involve building mediators to integrate systems in a purchase order and payment management scenario. Three more discovery scenarios target the automated discovery and invocation of suitable service providers for given specific service needs. While the testbed is small in terms of number of services (around a dozen), strong emphasis is put on providing realistic and detailed scenarios.

The SWS Challenge is organized as a series of workshops. Participants in the evaluation try modeling and solving the publicly available problem scenarios with their SWS technology. They are required to present and demonstrate their solution at a SWS Challenge workshop. A validation of the solutions is performed by teams

[1] `http://sws-challenge.org`
[2] `http://knowledgeweb.semanticweb.org`

composed of the workshop organizers and the peer participants. The validation results are published at the SWS Challenge website. The evaluation approach focuses on validating the functional coverage of an approach by certifying whether a particular aspect of a problem scenario was solved correctly or not. The general intention is to focus on whether a problem aspect was solved and how, that is the concrete techniques and descriptions an approach uses for the solution. The time a solution requires for execution or similar quantitative measurements are not within the scope of the SWS Challenge. Several evaluation approaches for measuring the flexibility of solutions and their software engineering benefits have been tried within the SWS Challenge. These will be discussed in detail in Section 8.6.4.

All but one scenario are accompanied by a testbed implementation of real SOAP based Web services. Solutions to the scenarios are required to properly interact with the testbed and the correct message exchange between the testbed and a scenario solution is verified. Furthermore, a peer code review is performed to promote the mutual understanding of each others approaches and to ensure that the solution works in the way claimed.

At the beginning of this thesis work, the SWS Challenge had just started. It provided two preliminary problem scenarios and an evaluation methodology was only just evolving. In fact, the originally defined methodology [VLZP06] has meanwhile been altered significantly, because it proved infeasible in some aspects. An established methodology how to reliably assess the functional scope of SWS technologies and a thorough understanding of the problem space as such, for instance about what are relevant problems, how they relate to each other and how they can be identified in a systematic way was lacking.

3.1.2. S3 Contest on Semantic Service Selection

The Contest S3 on Semantic Service Selection[3] is an annual international contest for the comparative performance evaluation of implemented SWS matchmakers. It was founded in 2006 at the 6th International Semantic Web Conference in Athens, GA, USA and is led by Prof. Klusch's group at DFKI Saarbrücken (Germany) in cooperation with people from France Telecom Research (France), SRI International (USA), NTT DoCoMo Research Europe (Germany), University of Zurich (Switzerland), University of Southampton (UK) and the candidate's research group at University of Jena (Germany). Its first three editions were hosted by the International Workshop on Service Matchmaking and Resource Retrieval in the Semantic Web[4] held in conjunction with the International Semantic Web Conferences in 2007, 2008 and 2009.

[3] `http://www.dfki.de/~klusch/s3/`
[4] `http://www-ags.dfki.uni-sb.de/~klusch/smr2/`

With the exception of recent extensions in 2009 which will be described in Chapter 7, the contest adopted a well-established evaluation approach from Information Retrieval (IR). As a basis for the evaluation test collections are provided. The collections contain a number of semantically annotated service offers, a (smaller) number of semantically annotated service requests and corresponding relevance judgments. These relevance judgments are provided by human experts and specify for each offer-request pair how relevant the offer is for the request, i.e., whether or to which degree the offer is able to satisfy the request.

For comparative performance evaluation, the S3 Contest readily provides the test collections OWLS-TC[5] for OWL-S and (since 2008) SAWSDL-TC[6] for SAWSDL services. The task to perform is to return for each given request a list of service offer descriptions semantically relevant to the given request ordered by estimated degree of relevance. Matchmaker implementations are then evaluated by comparing their output rankings with the one induced by the relevance judgments. The measures employed are classical retrieval effectiveness measures from IR based on recall and precision (see Chapter 7 or [BYRN99]). Furthermore, runtime performance is measured in terms of the average query response time, the aggregated runtime to match the complete test collection and the memory consumption during the matchmaking. This general procedure is largely agreed upon. However, there are several problems.

First, there has been almost no work that investigates the influence of different settings on the stability and meaningfulness of the evaluation results. For instance, relevance judgments can be binary or graded, the number of relevance judges and the resolution of conflicting judgments can vary, the characteristics of the test data being used may differ, various evaluation measures in different parameterization can be used to actually measure the quality of a produced output ranking, etc. Informed decisions about the evaluation measures employed, the underlying model of relevance, the procedure of how to obtain reliable relevance judgments, etc. are necessary for meaningful evaluations but impossible without a thorough understanding of the associated problems and their influence on the evaluation.

Second, in the area of SWS, there is not yet a standard how to describe the semantics of services. There is no consensus about the modeling approach to follow (e.g., WSMO, OWL-S, SAWSDL, ...) and also not about the semantic formalism to employ (WSML, OWL, Description Logics, Logic Programming languages, First Order Logic, ...). The approach of the S3 Contest requires providing readily usable test collections of semantic service descriptions. Thus, the evaluation presupposes certain formalisms (currently OWL-S and SAWSDL) and is limited to matchmakers which are able to process this formalism. It does not allow evaluating matchmakers

[5] http://projects.semwebcentral.org/projects/owls-tc/
[6] http://projects.semwebcentral.org/projects/sawsdl-tc/

based on other formalisms or comparing the performance of matchmakers across formalisms.

Third, defining meaningful ways how to describe the semantics of services, how to model a domain of discourse and how to choose the appropriate level of detail to formalize is one of the core research problems in the area. By providing the semantic descriptions used for an evaluation, the S3 Contest makes particular choices with respect to these questions. This poses difficulties to making the semantic annotations themselves subject of the evaluation.

Fourth, on a more practical level, an evaluation approach like that of the S3 Contest heavily depends — even more so than other approaches — upon the quality of the provided test data. However, the provided test collections have several limitations. For one thing, they are limited with respect to their actual use of the underlying description formalism. The OWLS-TC descriptions are, among other things, limited to modeling inputs and outputs of services without modeling preconditions or effects. The SAWSDL-TC descriptions are limited to one single interface and one single operation per interface and to using model references pointing to concepts described in OWL-DL exclusively. Both collections generally make very limited use of the semantic expressivity offered by the employed formalisms. For another thing, some of the services in OWLS-TC and SAWSDL-TC are generally viewed as somewhat idiosyncratic and unrealistic. Furthermore, in many cases it seems that services and requests have not been developed independently of each other. All these issues may reduce the real world relevance of an evaluation based upon these collections [KLKR07]. This will be covered in more detail in Chapter 5.

The problems of OWLS-TC and SAWSDL-TC have been generally acknowledged by the organizers of the S3 Contest. In fact, neither collection claims to be a standard test collection and the OWLS-TC manual states explicitly:

> "OWLS-TC2 has been specifically designed to be balanced with respect to the matching filters of the OWLS-MX, a hybrid semantic service matchmaker, developed at the German Research Center for Artificial Intelligence (DFKI). [...] Please note that no standard test collection for OWL-S service retrieval does exist yet. As a consequence, OWLS-TC can only be considered as one possible starting point for any activity towards achieving such a standard collection by the community as a whole." [KFK07]

The lack of rich and large public semantic service collections is widely viewed as a critical problem of the community. When the S3 Contest was first proposed during a meeting at the 2006 International Semantic Web Conference in Athens, GA, USA it was hoped and expected by the organizers that better collections would emerge from the community. Despite of corresponding efforts by the S3 Contest organizers this has unfortunately not been the case so far.

3.1.3. Web Service Challenge

The annual IEEE Web Service Challenge [BTW05, BCJW06, BCJW07, BBK+08, BWG09] is a third community evaluation initiative in the area of (semantic) Web services. It is held annually since 2005, but its focus has shifted considerably over the years.

It was started in 2005 with a design and two functional evaluations[7] [BTW05]. Unfortunately, no information about the design evaluation is available. In fact, detailed information about the setup, participants and results from this edition are generally not available anymore. As of December 2009 the website at `http://cssun.georgetown.edu/~blakeb/EEE05/` is not online anymore and communication with the organizers of the Challenge indicated that none of them still possesses the data from that edition.

The functional evaluation consisted of a service matchmaking and a composition challenge both based on the string equivalence of WSDL part names. Participants were provided with a directory of WSDL Web services specifications. Furthermore, each participant was given a specific discovery request as represented by provided input messages and the required output messages. For the matchmaking challenge, the participating systems had to output a list of services that either met or exceeded the requirements (did not use unavailable inputs and provided at least all requested outputs) and were evaluated based on the correctness and completeness of this list as well as the speed of discovery. The composition challenge worked in a similar way, but participating systems had to compute a sequence of services that met the specified IO requirements if not single service was able to fulfill them alone. This challenge was evaluated by the number of correctly composed services (shorter compositions were preferred) in addition to the speed of the composition process.

The 2006 edition[8] extended the previous setup by a track on semantic matchmaking and composition based on the compatibility of XML schema types [BCJW06]. The semantic track worked very much the same way like the syntactic track. However, instead of matching parameters based on WSDL part names, matching had to be performed based on XML Schema type inheritance. Unfortunately, results are not available on the 2006 edition website and again, communication with the organizers of the Challenge indicated that the corresponding data was not preserved.

In 2007, the syntactic tracks were discontinued to solely focus on semantic composition [BCJW07]. The setup of the other track remained largely unchanged except that OWL representations of the XML Schema data model were provided. Participants could choose whether to work with the XML Schema or OWL representation of the same data model (this of course implies that the usage of OWL was restricted to the expressivity also available in XML Schema). Unfortunately, the 2007 edition

[7] `http://www.comp.hkbu.edu.hk/~eee05/contest/`
[8] `http://www.vf.utwente.nl/~cec06/`

website[9] links to `http://www.ws-challenge.org` for all details about that edition including the results. As of December 2009 this URL is not available anymore.

The 2008 edition further evolved the Challenge by discontinuing the discovery track and focusing solely on service composition [BBK+08]. Furthermore, the XML Schema representation of the data taxonomy was also abandoned in favor of the OWL representation. Nevertheless the usage of OWL constructs was still restricted to simple inheritance. Other changes included the introduction of WSDL as query format and WSBPEL (Web Services Business Process Execution Language) as output format for the created compositions. Furthermore, exploitation of parallelism in the computed compositions was explicitly promoted. The evaluation was based upon the speed of the composition process, its completeness (number of valid compositions discovered), composition length (the shortest composition) and composition efficiency (usage of parallelism). The 2008 data and results are available at the 2008 edition website[10].

In 2009 quality of service information in form of response time and throughput was additionally provided via WSLA (Web Service Level Agreements) specifications for all component services [KBB+09]. Instead of finding the shortest composition, participants were challenged to find the composition with the lowest response time and highest throughput. Note that this implicitly requires participants to make use of parallelism wherever possible. Like in previous years there was also an award for the best solution architecture but the concrete judgment criteria for this award are not specified. All results from the 2009 challenge are available online[11].

The evolution of the WS Challenge over the years illustrates that it moved from a purely syntactic based challenge to incorporating more and more semantics. However, the semantics used by the challenge — basic type inheritance — are still much less expressive than usually employed in SWS frameworks. Furthermore, semantic descriptions do not include service categories, pre- or postconditions, but are restricted to input and output parameters.

Besides, the WS Challenge problems are stated in a way that there are unambiguous correct solutions. Given the problem statement that inputs must be available and outputs must be delivered, the computation of correct solutions based upon the provided type inheritance hierarchy is not conceptually difficult. The challenge of the WS Challenge is not to properly reason over given complex explicit or implicit semantics (like in case of the S3 Contest) or to formalize and process a rather difficult problem domain in the most suitable way (like in case of the SWS Challenge) but rather to traverse a very large search space in a smart and thus efficient way. I.e., the WS Challenge is really more of a speed performance than semantic

[9] `http://ws-challenge.georgetown.edu/wsc07/`
[10] `http://cec2008.cs.georgetown.edu/wsc08/`
[11] `http://ws-challenge.georgetown.edu/wsc09/`

challenge. This is also illustrated by the fact that the set of participants of the WS Challenge and the other two community evaluation initiatives is entirely disjoint.

Another interesting difference to the SWS Challenge and the S3 Contest is that these two initiatives rely on manually crafted test data. In contrast, the WS Challenge provides an automated data generator, capable of generating arbitrarily large testbeds [BWG09]. While this is a very reasonable approach if the evaluation focus is on computational speed, it is not very suitable if the evaluation focus is on semantics where a significant part of the problem always results from choosing how to formalize a real problem domain.

3.1.4. Project-Based SWS Evaluations

After having introduced to the three community evaluation initiatives in the area, we will now discuss some representatives of the project based evaluations in the field. Since evaluations are typically performed at the end of a project, we include only completed projects in our coverage. We furthermore focus on evaluations of core SWS tasks like service discovery, mediation or composition and do not deal in depth with related areas like ontology or general reasoning evaluation.

The project coverage contains primarily projects funded by the European Commission under Framework Program 6 (FP 6). For one reason, most funding for the field was provided by the EU. For another reason, EU projects make their deliverables available on the project websites. This is not always the case for other project types[12]. The coverage of projects is clearly not exhaustive but representative for the area.

DIANE – Services in Ad-hoc Networks

DIANE was a four year German Research Community (DFG) funded project of the candidate's research group at University of Jena which ran from 2002 till 2006: "The project aims at developing and evaluating concepts that allow for an integrated, efficient, and effective use of resources in the form of services in ad hoc networks. [...] In order to attain these goals, we propose mechanisms for the description, discovery and execution of services. Furthermore, we take a closer look at means of stimulating the provision of services."[13]

Within the DIANE project, a service description language (called DSD) and an accompanying middleware supporting service discovery, composition, and invocation

[12] No deliverables are publicly available for the SCALLOPS project (within which the OWLS-TC test collection was developed, `http://dfki.uni-sb.de/~klusch/scallops`) or the METEOR-S project (which provided significant input to the SAWSDL standard, `http://lsdis.cs.uga.edu/projects/meteor-s/`), for instance.

[13] `http://fusion.cs.uni-jena.de/DIANE/`

have been developed. This work has been evaluated within the context of the PhD thesis by Michael Klein [Kle06].

The DIANE evaluation focuses on four criteria an evaluation should measure:

1. *Degree of Automation:* Are the language and the tools powerful enough to allow for automatic and correct service usage? That means: Given a service request and service offers, will the discovery mechanism find the best-matching service offer and will it be possible to automatically invoke the service based on these results?

2. *Correctness of Matchmaking:* Is it possible to efficiently and correctly compute the correspondence of arbitrary offers and requests?

3. *Expressiveness:* Is it possible to describe real services and real service requests in sufficient detail to meet Criteria 1? Can this be done with reasonable effort?

4. *Decoupling:* Will a discovery mechanism be able to determine similarity between service offers and requests that are developed independently of each other? In other words: If a service requester writes his request without knowledge of the existing service descriptions, does the language offer enough guidance to ensure that suitable and only suitable services will be found by the discovery mechanism?

The DIANE evaluation has been performed in two parts. The first part, called inner evaluation, focuses on the evaluation of Criteria 1 and 2. These criteria were evaluated using a proof of concept implementation that demonstrated the feasibility of the developed approach based upon a fictitious scenario. Furthermore, the runtime performance of the proof of concept implementation was measured using artificially generated test data. Both is a rather common approach for project internal evaluations.

The second part, called outer evaluation, links the results of the inner evaluation to the real world by critically examining how well realistic services can be captured using DSD. Unlike most other projects, the DIANE evaluation aimed to design a reusable benchmark for this task. The benchmark is available online[14] and described in [Fis05].

It focuses on three aspects. The *effort* required to use the framework is assessed by measuring the initial effort to model the necessary ontologies as well as the continuous effort to maintain and update ontologies and service descriptions with the framework. The *expressiveness and correctness* of the framework is evaluated by assessing how well the semantics of given services can be captured by descriptions based on the employed formalism. Finally, the *level of decoupling* is evaluated by

[14] `http://fusion.cs.uni-jena.de/DIANE/benchmark/`

determining to which degree the framework still yields correct results, if services and goals are formalized by different people in a completely decoupled way. A detailed discussion of the concrete evaluation approaches to these three aspects will be provided later in this thesis in Section 4.3.

ASG – Adaptive Services Grid

ASG was a thirty month EU FP 6 project which ran from 2004 till 2007: "The Adaptive Services Grid (ASG) approach for semantic service provisioning is a solution to implement agility and adaptiveness promised by Service-oriented Architectures. Based on available standards, a solution for the complete service provisioning lifecycle has been built. A key concept of this solution is the usage of semantic information about services to automatically plan, enact, and monitor service compositions to fulfill user requests."[15]

Within the project, a survey of state of the art service and resource matchmaking has been performed [TIT+05][16], [TIR+07]. It presents a framework for the evaluation of semantic matchmaking frameworks by identifying different aspects of such frameworks that should be evaluated: query and advertising language, scalability, reasoning support, matchmaking versus brokering and mediation support. The survey analyzes and discusses a number of frameworks in the service as well as the grid community with respect to these criteria. However, the focus of the work is rather on the survey than on the comparison framework itself. While the framework does provide guidance for a structured comparison, it does not offer concrete test suites, measures, benchmarks or procedures for an objective comparative evaluation.

The technical deliverables do not contain extensive evaluations. Deliverable D2.I-4, for instance, presents the service matchmaker and query processor component developed within the project [ING+06, INS07]. It identifies several functional (e.g., describe and locate services) as well as non-functional (accuracy, performance and scalability, dependability) quality characteristics but does not present means to objectively assess those criteria.

Deliverable D7.IV investigates the business and market potential of the developed platform and its enabling technologies and concepts by discussing the business benefits of ASG with two use case scenarios [VTe06]. The concrete evaluation is performed by measuring the effort (time) for developing one of the use case scenarios within the proposed SWS framework. The benefit of ASG is then assessed by comparing that effort with the estimated one for a conventional development process without the support of the SWS framework. Unlike in the requirement analysis

[15] `http://asg-platform.org`

[16] This deliverable is not available on the project website, but it can be found at `http://www.sti-innsbruck.at/fileadmin/documents/deliverables/ASG/D2.I-2.pdf`

phase during the start of the project (see above) no comparison with alternative semantic service approaches is provided.

Knowledge Web

Knowledge Web was a four year EU FP 6 Network of Excellence project which ran from 2004 till 2007 with the objective "to coordinate the European research effort to make Semantic Web and Semantic Web Services a reality"[17].

Work package 2.4 of the project dealt with SWS, including the definitions of requirements and semantics for dynamic Web service discovery and automatic composition, invocation and interoperation. No dedicated evaluation deliverable is available. Within a deliverable about the "Architecture and Execution Semantics for the SWS" [VMZ+07] an approach to SWS is presented but the evaluation section only refers to the successful participation in the SWS Challenge without providing further details about this. Other deliverables like on "Semantics for Web Service Discovery and Composition" [LBC+05] and "Data Mediation in Semantic Web Services" [MSCV06] do not contain an evaluation section.

Deliverable D2.1.4 "Specification of a methodology, general criteria, and benchmark suites for benchmarking ontology tools" [GC05] contains a section on "Benchmarking semantic web service technology" which identifies candidate tools and names evaluation criteria (scalability, robustness, interoperability and usability) and plans that "different benchmark suites will be developed for benchmarking semantic web technology in order to assess the performance of these tools and the interoperability between the different types of tools." [GC05]. Unfortunately, it does not seem as if these plans have been fully pursued. However, the Semantic Web Services Challenge was founded as a Knowledge Web activity and supported by Knowledge Web until 2007 [VLZP06].

DIP Project – Data, Information, and Process Integration with Semantic Web Services

DIP was a three year EU FP 6 project which ran from 2004 till 2006: "DIP's objective has been to develop and extend Semantic Web and Web Service technologies in order to produce a new technology infrastructure for Semantic Web Services (SWS) — an environment in which different web services can discover and cooperate with each other automatically. DIP's long term vision is to deliver the enormous potential benefits of Semantic Web Services to e-Work and e-Commerce."[18]

Essential parts of the work on WSMO and WSMX were performed as part of DIP and several architectural specifications and prototypes resulted from the

[17] http://knowledgeweb.semanticweb.org
[18] http://dip.semanticweb.org/

project, among them prototypes for service discovery and composition. However, no dedicated evaluation or assessment deliverable is publicly available. Assessment of the developed prototypes is provided by means of case studies that motivate the need for SWS and demonstrate the feasibility of the developed approaches [DFLO06b, DFLO06a, WDR+04, GTD+07, Ric07]. One of the case studies has an associated evaluation of the resulting SWS enabled application, however, the corresponding deliverable is classified as confidential [Uns06].

SUPER – Semantics Utilised for Process management within and between Enterprises

SUPER was a three year EU FP 6 project which ran from 2006 till 2009: "The major objective of SUPER was to raise Business Process Management (BPM) to the business level, where it belongs, from the IT level where it mostly resides now. This resulted in development of tools enabling deployment of Semantic Business Process Management." In order to achieve this, SUPER aimed "to create the technological framework constituting BPM enriched with machine readable semantics by employing Semantic Web and Semantic Web Services accompanied by universal reference implementation for mechanized BPM"[19]. SUPER made most of the developed tools and ontologies (except for some which are rated confidential) available for download on the project website.

Among the contributions of SUPER is the YATOSP framework (Yet Another Telecommunications Ontologies Services and Processes), which aims to provide a reference for the creation of semantic business processes within the telecommunication sector [LCMdF+08]. Unfortunately, YATOSP is rated confidential. However, there is a comprehensive evaluation and assessment of YATOSP [LCMdF+08, dFEJ+09]. The deliverables present a set of requirements defined from a business perspective, qualitative and quantitative metrics for those requirements, test cases to measure those metrics and an analysis of the test results. The tests concern the YATOSP ontologies as well as associated development and management tools: "The metrics [...] refer to the quality aspects of YATOSP usage. They should also depict a satisfaction of using SUPER tools. The two aspects of satisfaction of using YATOSP and tools are inseparable" [dFEJ+09].

While the evaluation and assessment provided in SUPER is more comprehensive than that of many other projects, its evaluation approach is hard to reuse without access to YATOSP. Some of the defined metrics could be transferred to similar use case studies, but basically, without access to YATOSP, a new use case would have to be entirely redefined and redeveloped in order to use the evaluation approach for a comparative evaluation of alternative technologies.

[19] http://www.ip-super.org/

SWING – Semantic Web Services Interoperability for Geospatial Decision Making

SWING was a three year EU FP 6 project which ran from 2006 till 2009: "SWING aims at deploying Semantic Web Service (SWS) technology in the geospatial domain. In particular, we address two major obstacles that must be overcome for SWS technology to be generally adopted, i.e. to reduce the complexity of creating semantic descriptions and to increase the number of semantically described services. [...] The objective of SWING is to provide an open, easy-to-use SWS framework of suitable ontologies and inference tools for annotation, discovery, composition, and invocation of geospatial web services."[20]

The SWING technology is primarily demonstrated by means of three use case scenarios [DHL+07]. Furthermore, there is an experience report available which aims to serve "as a documentation of the success of the SWING project" [UB09]. However, this report provides rather a collection of informal feedback and lessons learned than a formal evaluation.

Besides, Deliverable 2.4, for instance, describes the realization of the service discovery engine within SWING [Hof08]. However, no evaluation is provided. The deliverable states: "As there is, to our knowledge, no similar discovery approach to the newly developed WPS [Web processing service] discovery methodology that we use in our prototype, we did not make any classical empirical performance comparisons" [Hof08].

Instead, the authors just compare their semantic approach with keyword based service discovery and finally conclude: "In empirical runs on the test cases considered in D3.2, the implemented prototype delivers the correct results and takes negligible runtime" [Hof08].

SIMS – Semantic Interfaces for Mobile Services

SIMS was a thirty month EU FP 6 project which ran from 2006 till 2008: "SIMS will provide tools for design and validation of service components. SIMS will provide middleware that enables discovery and validation of service opportunities between peers in ad-hoc interactions, and efficient deployment of service components through runtime composition of applications from service components. By making it possible to discover service component interoperability at runtime, SIMS will enable a new model for rapid deployment and delivery of reliable services."[21]

SIMS provides a detailed evaluation and assessment plan [SC07] as well as an evaluation of the SIMS approach [Shi08b] by means of demonstration via a number of trial services [Shi08a]. The evaluation and assessment plan remarks:

[20] http://www.swing-project.org/
[21] http://www.ist-sims.org/

> "A comparative approach, including the qualitative or quantitative comparison of various key process indicators in parallel projects, would yield the most valid results. However, such an assessment is outside the scope and means of SIMS. The project will evaluate the usability and basic assumptions of the SIMS approach, rather than proving its superiority compared to traditional approaches." [SC07]

The actual evaluation assesses how much time was spent using the various provided tools when implementing the trial service scenario. It then discusses the usability and perceived advantages and disadvantages of the different artifacts of the proposed development methodology, the implemented tools and the execution middleware.

CASCOM – Context-Aware Business Application Service Co-ordination in Mobile Computing Environments

CASCOM was a three year EU FP 6 project which ran from 2004 till 2007: "The main objective of the project (CASCOM) is to implement, validate, and trial a value-added supportive infrastructure for Semantic Web based business application services across mobile and fixed networks."[22]

CASCOM developed an infrastructure for dynamic service discovery and composition in peer-to-peer networks and included a work package on validation and trials [FSM+07, FMS+07]. Evaluation is based on three pillars: a field trial for evaluating whether the developed coordination framework meets the business needs, a usability lab trial for evaluating the usability of the system and a simulation/emulation part for evaluating non-functional characteristics of the CASCOM system. The field and usability lab trial are based on a healthcare emergency assistance scenario. The architecture and prototype are evaluated by letting users perform typical tasks and then evaluating their satisfaction via questionnaires.

The simulation/emulation part assessed the runtime performance and retrieval correctness of the service matchmaking component. The test setup largely corresponds to that of the S3 Contest described above and primarily used the OWLS-TC test collection also used in the S3 Contest. Furthermore the simulation/emulation comprised an evaluation of the runtime performance of the service composition planner, the service execution agent and a federated service directory, albeit without providing much detail about the used data.

For the two available composition planners, an additional evaluation is provided. The first one, called OWLS-XPlan, is evaluated using a publicly available benchmark of the international planning competition IPC3[23]. This benchmark allows

[22] http://www.ist-cascom.org/
[23] http://planning.cis.strath.ac.uk/competition/

measuring planning performance in terms of the planning completeness (total percentage of solved problems), the average plan length and the average plan quality (average distance of individual plans from the optimal plan length) in relation to the complexity of the given problems. The second planner, called SAPA, was evaluated with respect to domain independence, performance and scalability through testing it in different domains and a set of OWL-S services from the online medicine selling domain, unfortunately, without providing details about the test data or making it available.

RW^2 – Reasoning with Web Services

RW^2 was a thirty month project funded by the Austrian Federal Ministry for Transport, Innovation and Technology which ran from 2005 till 2007: "The RW^2 project follows the direction of research that describes Web Services using semantic annotations. The main objectives are: [...] A logic-based service discovery framework which allows agents to find suitable services based on a declarative specification of their needs as well as all other (non-logic based) elements used in the description of Web Services like Ontologies. [...]"[24]

A conceptual model of the discovery framework and an implementation based on a lightweight set-based modeling approach using WSML-DL is presented in Deliverable 2.2. The deliverable concludes: "Future work will be to find a set of suitable restrictions allowing also to use other WSML variants then DL to describe the services and goals. Moreover we plan to implement the more complex state based view on services that is outlined in the first chapter of this deliverable. Both discovery strategies still require more evaluation based on use cases. We plan to use the testbed provided by the Semantic Web Service Challenge to do so" [LKF06].

However, it does not appear as if these plans have been pursued. Except for illustrating use cases, no evaluation is publicly available. The project has created a reusable benchmark for the performance of logic reasoners which assesses their runtime performance for various reasoning tasks which will be covered in Section 3.2 [SKFL07]. Unfortunately, this benchmark does not allow deriving any direct conclusions about the performance of the discovery framework developed by the project.

Summary of Project-Based SWS Evaluations

There is great variety in how results are presented and organized within the projects. Some projects include a distinguished evaluation or assessment work package or deliverable making it very easy to access the desired information (CASCOM and SIMS), others perform a more or less comprehensive evaluation, but do not mark it

[24] `http://rw2.deri.at`

very prominently (SUPER and to a lesser degree ASG), for most we were not able to identify a fairly comprehensive publicly available evaluation or assessment of the technical contributions of interest within the scope of this thesis (Knowledge Web, RW^2, DIP and SWING).

Generally, the amount of information provided especially by the large projects is rather overwhelming, typically consisting of several thousands pages of paper in dozens of deliverables. In the absence of an established standard structure that makes it easy to identify the relevant information it is extremely hard to dig down to the proper deliverables. It may be that we missed or misinterpreted relevant information in some cases. Furthermore, our survey was restricted to publicly available information. Thus, the summary statements provided below need to be taken with some caution and certainly do not imply a judgment of the project contributions or success.

While there are some exceptions, the overall standard of evaluation and assessment in the area is clearly not optimal. This confirms the discussion from Section 1.2. Evaluation and assessment generally receives a very minor share of the total project effort. Of the mentioned projects, only DIANE and Knowledge Web aimed at creating reusable benchmarks. Both will be discussed in more detail in Section 4.3.

The other projects demonstrated the feasibility of the developed approaches via implementations of use case scenarios. SUPER, SIMS, CASCOM and ASG additionally performed an assessment of the user satisfaction with the proposed methodologies and frameworks. Such evaluation aims at showcasing the benefit of using semantics on top of conventional Web service technology and at assessing the usability of the developed tools. Unfortunately, assessment approaches are very use case and tool specific. They may serve as input to the definition of generally usable benchmarks, but they can not serve as such as is without full access to the use cases and tools. Generally, the available information is also not structured to support reusability. In case of SUPER and DIP, essential information is rated confidential and thus not available. None of the projects provided a significant number of semantically described services that could be used as a base of experimental comparison.

In total, project internal evaluations and assessments are not designed in a way that makes them easily transferable to other projects or allows comparison of different approaches. Furthermore, none of the surveyed projects performed an explicit comparative evaluation of the project contributions to other approaches to SWS, much less an experimental or quantative one. The corresponding statement from the evaluation performed within the SIMS project properly summarizes the approach taken by more or less all projects:

> "A comparative approach, including the qualitative or quantitative comparison of various key process indicators in parallel projects, would yield

> the most valid results. However, such an assessment is outside the scope and means of SIMS. The project will evaluate the usability and basic assumptions of the SIMS approach, rather than proving its superiority compared to traditional approaches." [SC07]

3.1.5. Other SWS Evaluation Efforts

We complement the coverage of the state of the art in SWS evaluations with a report of evaluation efforts besides the mentioned community initiatives and evaluations performed within projects in the area.

SEALS – Semantic Evaluation At Large Scale

Just recently, the EU funded SEALS project has started in June 2009. The goal of SEALS is to provide means for automated benchmarking in the areas of ontology engineering tools, storage and reasoning systems, matching tools, semantic search tools and semantic Web service tools: "The SEALS project will create a lasting reference infrastructure for semantic technology evaluation (the SEALS Platform) and thus facilitate the continuous evaluation of semantic technologies at a large scale. The SEALS Platform will be an independent, open, scalable, extensible and sustainable infrastructure that will allow the evaluation of semantic technologies by providing an integrated set of evaluation services and test suites."[25] However, as of the time of writing of this thesis (December 2009), no details on the evaluation data sets and measures were available yet.

An Evaluation Platform for Semantic Web Technology

In [ÅÅLS06] and her PhD thesis [Åbe07], Åberg proposes a platform to evaluate service discovery in the semantic web. However, her platform is rather a software architecture to provide some guidance in the development of SWS frameworks than an evaluation platform and it does not become clear how this platform can help to comparatively evaluate different Web service frameworks. She also proposes a test suite for service discovery evaluation composed of five service requests which she uses to evaluate OWL-S, WSMO, and OWL-DTP (a service description language proposed in her thesis): "The evaluation consists of attempting to use each approach to express each query and analyze whether or not it is possible and whether there are difficulties with respect to matching the expected services" [Åbe07].

Each service request is used to highlight one functional challenge associated with semantic service discovery:

[25] `http://seals-project.eu`

1. Describing a needed resource (book a flight from Monterey airport to JFK airport).
2. Specifying the conditions on the use of the service (get a bed delivered within a week).
3. Specifying the kind of business transaction provided by the service (get a research article for free).
4. Specifying conditions on the means used by the provider to provide the service (get a t-shirt with the assurance that it has not been produced through child labor).
5. Specifying constraints on the service provider (get a football match ticket from a trusted provider).

While these challenges and the corresponding discussion of OWL-S, WSMO and OWL-DTP form an interesting initial starting point for evaluation, she does not provide further tests or evaluation measures or methods. She also does not establish a relation between the test suite and the proposed evaluation platform or between these two contributions and the related work (including all approaches described above).

On the Evaluation of Semantic Web Service Matchmaking Systems

Tsetsos et al. published on the evaluation of SWS matchmakers [TAH06]. They were the first to identify and discuss problems around the state of the art in SWS matchmaking evaluations. Their interest primarily concerned the proper evaluation of imprecise matches and they suggested the use of fuzzy evaluation measures to address this issue. To the best of our knowledge, their corresponding work from 2006 has not been continued. It will be discussed in depth in Chapter 7, to which it is directly relevant.

A Framework for the Evaluation of Semantics-based Service Composition Approaches

Very recently (2009), Silva et al. published a discussion about the evaluation of SWS composition approaches [SPvS09]. They discuss issues around the system architecture being evaluated, the used data, scenarios and metrics and identify some problems with this respect. In particular, they deal with issues around obtaining suitable test data (which will be covered in depth in Chapter 5) and evaluation measures for automated service composition. This work is most relevant to the above described Web Service Challenge. However, quantitative measures for automated service composition are not within the scope of this thesis (see Section 3.3).

Verification of Results in Scientific Papers

In addition to the work covered above, the validation of more specific technical contributions presented in the various publications in the field is obviously also relevant to this thesis. While many approaches are presented without a proper validation, some papers present thorough evaluations. Klusch et al., for instance, present detailed experimental evaluations of two proposed semantic service discovery algorithms in terms of runtime performance and retrieval correctness [KKF08, KK07].

However, we are not aware of any publications specifically on generally usable testbeds and evaluation methodologies or comparative evaluations of different technical SWS approaches apart from the ones listed above. Where applicable, we will cover related work from validations of technical papers in the chapters of this thesis to which they are most related. The work by Klusch and colleagues, for instance, will be covered in Section 7.10.

3.2. Benchmarking and Evaluation in Related Areas

Apart from the work which is devoted to the evaluation of SWS directly and has been described in the previous section, benchmarking and evaluation plays an important role in related areas, too. In this section a brief introduction to evaluation efforts in the areas directly related to this thesis will be provided.

Software Evaluation

Semantic Web Services in general can be considered a novel way of developing and maintaining distributed computer systems. In a broad sense, they are thus an application of software engineering. Therefore, this thesis also builds on the work on evaluation of software systems from software engineering. Since the 1990's, evaluation, measurement and experimentation has received increased interest in this community.

Wohlin et al. provide an introduction to experimentation in software engineering [WRH$^+$00]. Fenton has authored a standard text book on software metrics [Fen91] as did Bache and Bazzana [BB93]. Another example for an excellent introduction to software measurement is the guidebook by Park et al. [PGF96]. All these textbooks provide classifications of evaluation approaches, guidelines for experimental design and introductions to quality assurance and measurement as well as measurement theory. More complete introductions to measurement theory are available for instance in [Fin84, Fin82]. Gediga et al. and Weiderman et al. are mentioned as other examples for literature on evaluation from software engineering [GHD02, WHBK87].

Several standards for software quality and evaluation are available from the International Organization for Standardization (ISO). The most important are ISO 9126 and ISO 14598. ISO 9126 deals with software product quality and defines a system of quality attributes for software (functionality, reliability, usability, efficiency, maintainability and portability), internal and external quality metrics as well as metrics for the quality in use of a complete software product. ISO 14598 deals with software product evaluation and standardizes the planning, management and execution of evaluations.

The work on evaluation, experimentation and measurement from software engineering has influenced the general design and approach of the benchmarks contributed by this thesis. Furthermore, the conceptual model for SWS evaluation presented in the following chapter is heavily based upon techniques developed in software engineering. An approach to the definition of measures from software engineering, for instance, is used to derive the criteria model for SWS evaluation from an analysis of the engineering goals motivating SWS.

On a more concrete level, however, most of the work on evaluation from software engineering is concerned with the evaluation of software with respect to the quality of an implementation, e.g., the maintainability or structure of the code or the usability of an interface. This thesis is concerned with SWS, a still emerging research area. Thus, it is less concerned with evaluating the strength and quality of implementations in the area. Instead, it is primarily concerned with evaluating the comparative strengths and weaknesses of approaches to specific technical problems on a more fundamental, basic level.

This is similar to evaluation and benchmarking of more specific problems in other areas of computer science, in particular those that have a rich history of standard benchmarks. Among these are the already mentioned SPEC benchmarks for processing performance, TPC for relational databases or TREC from IR. Chapter 7 relies heavily on contributions from TREC in particular and the IR community in general. The related work from this area will thus be reported there. Apart from that, benchmarks like TPC or SPEC inspired the general benchmarking approach, but are not directly related to this thesis.

More closely related areas are ontology evaluation, ontology alignment evaluation, benchmarking of reasoners and evaluation of planning from AI. In the following, references to standard work in these areas will be provided and the relationship to this thesis will be defined.

Ontology Evaluation

An overview of the work on ontology evaluation has been provided by Gómez-Pérez [GP04]. Evaluation criteria (consistency, completeness, conciseness, expandability and sensitiveness) are introduced, metrics for measuring and assessing them

are presented and an overview of tools supporting the evaluation of ontologies is given. A more recent work on defining metrics for ontology evaluation and automating the evaluation process is available in [VS07, VVSH08]. Within the scope of this thesis, ontologies will be evaluated from a functional point of view only. I.e., within the scope of this thesis, the influence that different ontology languages, modeling styles and formalization approaches have on the effectiveness of algorithms operating on semantic descriptions based upon ontologies will be of interest. However, the evaluation of intrinsic quality attributes of ontologies is beyond the scope of this thesis.

Ontology Alignment

Ontology matching or alignment refers to the problem of finding correspondences in different formalizations of related problem domains. Ontology alignment is closely related to data mediation in the area of web services and in particular involved whenever different unaligned ontologies are referenced by different SWS descriptions being processed. A comprehensive coverage of the topic of ontology matching is provided in [ES07], including a chapter on evaluation of matching systems. In many ways the evaluation of ontology matching is similar to the evaluation of SWS related tasks. Like semantic service discovery, for instance, the evaluation of ontology alignment is based upon techniques from Information Retrieval, which needs to be extended and adapted in order to be suitable for this particular problem domain. This will be dealt with in Section 7.3. While the problems in evaluation of ontology matching and SWS technologies are similar, they require different solutions. Among the most important differences is the existence of standard representation languages for ontologies (e.g., RDF and OWL) which simplifies evaluation of ontology matching algorithms. In contrast, in SWS there are no established standard formalisms. This is one of the central problems that adds a layer of complexity to the evaluation of SWS technologies.

Reasoning and Triple Stores

Reasoning refers to the process of applying logical inference rules for deducing implicit knowledge from a given formal knowledge base. Logic reasoning is involved in SWS related tasks whenever logic formalisms are used to describe the services being processed. Reasoners like KAON2[26], RACER[27] or Flora2[28] are frameworks processing ontologies and offering various reasoning tasks like subsumption classification or consistency checking for those ontologies. Closely related to these are

[26] http://kaon2.semanticweb.org/
[27] http://www.sts.tu-harburg.de/~r.f.moeller/racer/
[28] http://flora.sourceforge.net/

RDF triple stores and SPARQL query processors. Evaluation of such frameworks is primarily performed with respect to correctness and in particular processing speed. Commonly used benchmarks in the area include the Lehigh University Benchmark (LUBM)[29] [GPH05], which has been extended by Ma et al. [MYQ+06] and Kolas [Kol08], the Berlin SPARQL Benchmark[30] [BS09] and the SP^{2}Bench[31] [SHLP09]. General issues around the topic are covered in [GCGP05, GQPH07]. While reasoning efficiency is often a key factor for SWS processing efficiency, runtime performance is not within the primary interests of the benchmarking contributions of this thesis. Furthermore, similar to the work on ontology matching evaluation, the work on reasoner benchmarking benefits from existing standard languages. Performance benchmarks are defined with respect to these standards, similar as benchmarks in the database community are defined with respect to the features offered by the SQL relational query language. As argued above, this is not yet possible in the area of SWS.

AI Planning

AI Planning has been an active research area for decades and recently gained new momentum through its usage as a technique to achieve (semi-) automated service composition [Pee05]. The International Planning Competition IPC "is a biennial event organized in the context of the International Conference on Planning and Scheduling (ICAPS), which has several goals, including analyzing and advancing the state-of-the-art in automated planning systems; providing new data sets to be used by the research community as benchmarks for evaluating different approaches to automated planning; emphasizing new research issues in planning; promoting the acceptance and applicability of planning technology."[32] IPC is based on the Planning Domain Definition Language (PDDL). The recent edition of the competition evaluates exclusively the quality of computed plans while previous editions have also evaluated the runtime performance of the planners. To the best of our knowledge no direct applications to the Web service domain are available. Besides, the evaluation of service composition is not within the core focus of this thesis (cf. Section 3.3).

A related event to the IPC is the Trading Agent Competition[33]. The trading agent problem deals with game like scenarios where autonomous agents try to mimic intelligent human behavior for optimizing their profit in competitive environments like the assembling and booking of travel packages via auctions. SWS are viewed as an

[29] http://swat.cse.lehigh.edu/projects/lubm/
[30] http://www4.wiwiss.fu-berlin.de/bizer/BerlinSPARQLBenchmark/
[31] http://dbis.informatik.uni-freiburg.de/index.php?project=SP2B
[32] http://ipc.informatik.uni-freiburg.de/
[33] http://www.sics.se/tac/

enabling technology for autonomous agents, but the focus of SWS and autonomous agents are slightly different. While SWS are primarily concerned with formalizing resources (data and services) to leverage their automatic consumption in open environments the focus of the trading agents competition lies on the implementation of intelligent strategies in well known and restricted domains.

3.3. Conclusions

Within this chapter the state of the art in the context of this thesis has been presented. In particular the coverage of project based evaluations highlights that standards for the evaluation of research contributions are largely lacking. Furthermore, we were unable to identify work specifically on the fundamentals of evaluation in the area of SWS. These fundamentals include how to choose the evaluation criteria, how to properly measure them, how to achieve reliability and validity of the evaluation procedure and how to assess evaluations themselves.

As is evident from the presented survey, project validations focus on use case centric evaluations that illustrate the feasibility of the developed approaches and, to a lesser degree, their advantage over traditional software engineering methods. Typically, these evaluations are difficult to reuse since they are specific to the evaluation use case for which details are not always available. Apart from two exceptions (the DIANE Benchmark and the Knowledge Web SWS Challenge activities) none of the surveyed projects aimed at developing reusable benchmarks that may evolve into standards. In fact, none of the projects compares the approach to SWS developed within that project with alternative approaches.

In contrast, the three community evaluation initiatives in the area of SWS strive at establishing such standards and providing such comparison. The SWS Challenge focuses on certifying the functional scope of SWS frameworks and promoting a deeper understanding for how the different approaches actually work. The S3 Contest strives at quantitatively evaluating the retrieval correctness and runtime performance of different service discovery approaches. Finally, the WS Challenge aims at evaluating the runtime performance of service composition approaches, albeit ones based on much lower level semantics than the other two initiatives.

However, all these initiatives were not yet available or in an initial stage when this thesis work started in 2005. The SWS Challenge was founded in 2005, the S3 Contest in 2006 and the WS Challenge covered only syntactic discovery and composition in 2005 and only later added some limited form of semantics. This thesis thus coevolved with these initiatives and essential parts of the thesis work were performed under the umbrelly or in collaboration with these initiatives.

The conceptual model for SWS technology evaluation which will be presented in the following Chapter 4 offers the previously missing treatment of the general

fundamentals of SWS evaluation. It is equally applicable to all SWS related tasks and contexts. The framework allows a structured analysis and comparison of SWS evaluation efforts. Such detailed analysis will be provided for the three community evaluation initiatives and the DIANE benchmark (i.e., the efforts striving at the development of reusable benchmarks) in Section 4.3. Based upon this analysis, the scope of this thesis will be further delineated and the choice of its concrete benchmarking contributions motivated. Additionally, the analysis will further relate these contributions to the state of the art.

These considerations conclude the first part of this thesis. The following second part presents the thesis' main contributions towards the evaluation of SWS technologies.

Part II.

Evaluation of Semantic Web Services Technology

CHAPTER 4

Conceptual Model for SWS Technology Evaluation

> Measurement presupposes something to be measured, and, unless we know what that something is, no measurement can have any significance.
>
> *(Peter Caws)*

To lay the foundation for a systematic approach to SWS evaluation, a conceptual model for SWS technology evaluation is presented. The first part of the model will be introduced in Section 4.1 where the possible criteria dimensions for SWS technology evaluation will be identified and motivated. This is complemented in Section 4.2 by a requirements catalogue to assess and meta-evaluate SWS technology evaluations. Finally, Section 4.3 reviews the state of the art as presented in Chapter 3 with respect to the presented conceptual model. Based upon this review the further contributions of this thesis are motivated and put into context. The work presented in this chapter has been partially published in [KKRPK08, KLKR07]

4.1. Criteria Dimension Model

The first important question related to any evaluation endeavor regards the criteria according to which the object of interest should be evaluated, i.e. which characteristics of the evaluand to investigate. As described in Chapter 3, different SWS evaluation endeavors have so far focused on very different criteria, mostly without motivating the choice of criteria or discussing how the different criteria relate to each other.

We chose the Goal-Question-Metric (GQM) approach from software engineering for deriving a well-justified set of evaluation dimensions. The GQM approach is introduced in the following. Subsequently, a SWS technology goal analysis is presented. Based upon this goal analysis, the GQM methodology is applied to derive the desired set of evaluation dimensions. This section concludes with a discussion of the derived criteria model. The presented work has been published in [KKRPK08].

4.1.1. Goal-Question-Metric Approach

The GQM paradigm is a mechanism for defining and evaluating a set of operational goals, using measurement [Bas92, BCR94]. Since being developed at NASA in 1984 it has become a recommended gold practice of the US Department of Defense Information Analysis Center[1] and has been used in various software engineering projects worldwide.

GQM is based on the assumption that the evaluation of any system should be an evaluation of fitness for purpose. Thus, any evaluation activity should be preceded by the identification of the engineering goals behind the system or technology to be evaluated. These goals then need to be traced to the data that is suitable to define those goals operationally and make them quantifiable and measureable.

A measurement model derived using the GQM paradigm distinguishes three conceptual levels:

Conceptual level - Goals Goals are defined for an object, for a variety of reasons, with respect to various models of quality, from various points of view, relative to particular environments. Objects of measurement can be products, processes, or resources.

Operational level - Questions A set of questions is used to characterize the way the achievement of a specific goal is going to be performed based on some characterizing model. Questions aim at characterizing the object of measurement with respect to a selected quality issue. They are suppposed to make the goals measureable by linking them with the data that needs to be obtained to quantify the achievement of the goals.

Quantitative level - Metrics A set of data is associated with every question in order to answer it in a quantitative way. The data can be objective, i.e. depending only on the object being measured, or subjective, i.e. depending on both the object being measured and the viewpoint from which the measurement is performed.

[1] `https://www.goldpractices.com/practices/gqm/`

The process of setting goals is critical to the successful application of the GQM approach. In order to derive a generally applicable evaluation model for SWS technology the engineering goals motivating SWS technology in various settings and projects have to be identified. To reflect an as broad as possible view on the issue, a literature study of use cases motivating SWS technology was performed. It will be presented in the following section.

4.1.2. SWS Technology Goal Analysis

The obvious overall goal of SWS technology is to increase productivity and efficiency by supporting or (partially) automating the process of consuming functionality offered as a service (see Section 2.3). However, the precise use case motivating particular approaches to SWS is often not clearly identified. To identify the main objectives motivating SWS, a review of published work with a focus on detailed and specific descriptions of envisioned use-cases was performed [MSZ01, OLES05, TRF06, AL05, PSK03, STR06, CLC+04, PCB+05, FN06, Pre07, KKRKS08, Pre04, STK+04, KKR07d, LH03, RSN+07, CNS+05, BLW04, FG05, GTD+06, KKRM05, AHKZ08]. While this review is clearly not exhaustive, it is representative for the majority of SWS projects. It was found that published SWS use cases can be roughly divided in two types of application domains.

The first type envisions enabling late dynamic service discovery, selection and binding at run-time, typically in the domain of e-Commerce. In mobile environments, the non-availability of stable services forces to discover and bind services dynamically (e.g. booking local attractions via mobile devices while traveling [TRF06, STR06, KKRM05]). In B2B scenarios, the dynamic and autonomous reaction to changes in the service landscape allows taking advantage of the appearance of better or less expensive services or recovering from failures by automatically replacing faulted or offline services [PSK03, CLC+04]. Many scenarios involve the dynamic selection of service instances based on similar re-appearing goal instances in B2B relationships: the location of suitable carriers to provide transportation services [PCB+05, FN06, KKRKS08], an intelligent procurement management for supplies [PSK03, Pre07, KKRKS08] or the location of the most apropriate notification service to contact a customer [CLC+04]. In B2C relationships, SWS are often motivated by the desire to delegate a search for the best among many options to autonomous agents. In these scenarios, many providers offer similar services and the best provider depends on the concrete goal or varies over time. Typical scenarios of this type involve the discovery of the best deal to purchase a set of items [Pre04, STK+04, KKR07d, LH03, RSN+07, AL05], to find the best matching offer in an appartment rental, dating, or job search scenario [CNS+05, NSD07], or to make travel arrangements and flight or hotel bookings [MSZ01, PSK03, BLW04].

The main focus in all of the above mentioned contexts is on discovery, matchmaking and precise filtering or ranking of many possible options. Usually a high degree of automation is sought, in some scenarios complete automation is required.

The second type of application scenarios deals with supporting developers in establishing or maintaining rather stable B2B or B2C relationships and setting up distributed applications. Such scenarios root in application domains like Business Process Management (BPM) and Enterprise Application Integration (EAI). In these fields, SWS are motivated by the desire to decrease the programming time and cost by semi-automating very time consuming tasks like the location of services in appropriate registries [AHKZ08] or the establishment of data and process mediation procedures [OLES05]. Scenarios in this category include the provision of value added services by bundling or mediating external contractors [FG05], the semi-automated design of processes to manage virtual ISP problems [Pre04], or the development of an emergency management system in the e-government domain [GTD+06]. The goal of employing SWS in such settings is to ease the process of integrating remote systems, master the encountered heterogeneity, and decrease the level of coupling between the components. Full automation is usually not required.

From the use cases listed above four main high-level goals of SWS technology are derived. Following the GQM approach these are further defined by a set of questions characterizing each goal in a measurable way. The goals and defining questions are described in the following, referenced with the use cases from which they were derived.

Goal 1 *Allow the dynamic and transparent usage of functionality in mobile or P2P environments where the availability and reliability of that desired functionality is not under local control [TRF06, STR06, KKRM05].*

1. Does the framework allow use of external functionality as if it were locally available? Is the framework able to hide the fact that the functionality is dynamically discovered and bound and supports full automation?
2. Does the framework guarantee correctness to allow for full automation?
3. If required, does the framework work under the requirements of P2P environments or the limited resources of mobile devices?

Goal 2 *Minimize the cost or optimize the quality of a consumed functionality by dynamically reacting to changes in the service landscape [MSZ01, PSK03, CLC+04, PCB+05, FN06, Pre07, KKRKS08, Pre04, STK+04, KKR07d, LH03, RSN+07, CNS+05, BLW04, FG05, KKRM05].*

4. Does the usage of the framework decrease the time necessary to find a good enough or the optimal option? To what extent?

5. Does the usage of the framework increase the quality of the option discovered? To what extent?

Goal 3 *Reduce failures or down-time by automatically replacing faulted or unavailable service components in a distributed application [PSK03, CLC+04, PCB+05, Pre07, KKRKS08, Pre04, FG05, GTD+06].*

6. Does the framework support to react autonomously to detected failures?
7. If a human still needs to be in the loop, to what extent does the framework support that human and reduces the time necessary to recover from failures?

Goal 4 *Ease the development and management of complex software systems by allowing the specification of programs at a higher level of abstraction and semi-automating development tasks necessary to compose services into applications [FG05, Pre04, GTD+06].*

8. Does the framework allow to (semi-)automate tasks that previously had to be performed manually? Which tasks and to what extent?
9. Does the framework decrease the effort required to reuse components in other contexts?
10. Does the framework lead to increased programmer productivity when creating, maintaining, or adapting service-oriented applications?

Furthermore, there are a number of cros cutting questions related to all goals:

11. How tightly coupled are service providers and consumers in the framework (e.g., do they need to use a common vocabulary)?
12. How much effort is it to use the framework, e.g. to publish service offers or formalize goals with the framework?
13. How much effort is it to set up and maintain the framework as such (e.g. agree on common ontologies if that is necessary)?
14. How correctly and completely does the framework work? Does it act like the user it acts on behalf of? How often does it fail to find a correct solution even though one exists? How often does it find an optimal solution? In such cases, how short of optimal is the solution provided by the framework?
15. How well does the framework scale?

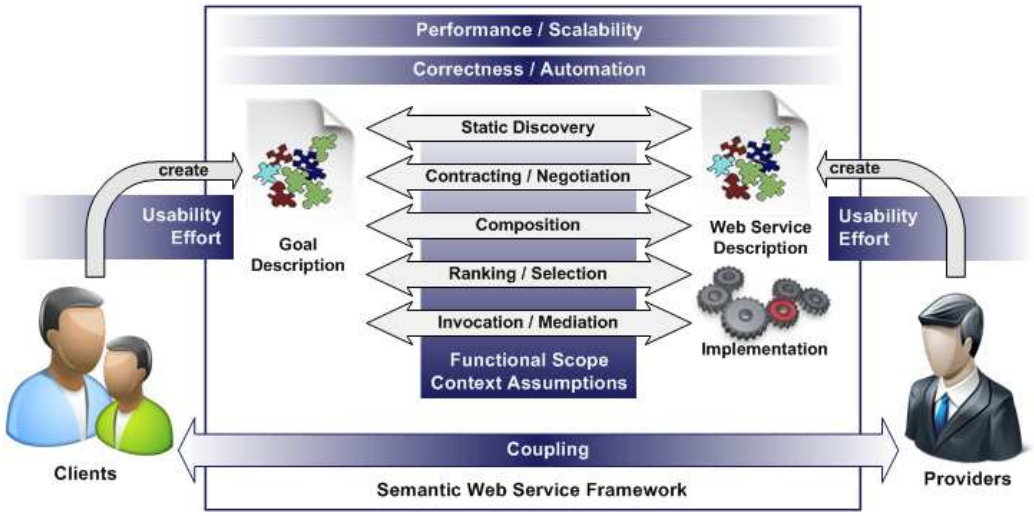

Figure 4.1.: Conceptual criteria model for the evaluation of SWS technology

Finally, it is essential to keep in mind that the performance of any framework will typically vary depending on given context parameters at hand and must not be easily generalized:

16. For which types of use cases concerning application types, service characteristics, business models etc. are the answers to the previous questions valid? How do the answers change in a changing context?

4.1.3. Derivation of Evaluation Dimensions

The questions characterizing the concrete software engineering goals motivating SWS technology in different use cases can now be used to derive a set of underlying high-level quality dimensions. An analysis of the correlations among the questions was performed to derive the conceptual criteria model for the evaluation of SWS frameworks displayed in Figure 4.1. This model comprises the following dimensions of evaluation.

Performance / Scalability regards the runtime performance and scalability characteristics of a framework. It is typically measured by the computing resources required (e.g. processor time or memory). Questions 3 and 15 are related to this dimension.

Usability / Effort regards the usability of the framework in terms of how much effort is required to set it up, maintain it, and use it. This dimension is

influenced for instance by the complexity of the framework and the available tool support. Questions 4, 7, 9, 10, 12 and 13 are related to this dimension.

Correctness / Automation regards the quality and extent of the support offered by the framework. Correctness relates to the degree to which a framework acts precisely like the user it acts on behalf of. Automation concerns the degree to which tasks previously performed by humans are now performed by the framework. A high degree of automation obviously requires a high level of correctness. Correctness is closely related to the often used notion of *expressivity* that captures how precisely and comprehensively a service's capabilities and a user's needs can be formalized in a framework. Question 1, 2, 4–8, 10 and 14 are related to this dimension.

Coupling regards how tightly coupled the providers and the consumers of services are in this framework, e.g., whether they have to agree on common ontologies or not. Questions 9, 11 and 13 are related to this dimension.

Functional Scope / Context Assumptions regards the functional scope of a framework, i.e., which of the tasks typically involved in using functionality offered as a service (discovery, negotiation, composition, data and process mediation etc.) are supported by the framework. This, as well as other assumptions regarding the type of services, the business model, the service usage model, the user characteristics etc. also defines the context for the other dimensions. Assessments made with respect to those will always depend on the context at hand. Questions 1, 8 and 16 are related to this dimension.

4.1.4. Discussion of the Criteria Model

Some remarks about the presented criteria model need to be made. First, designing a criteria model involves some degree of freedom how to design it. We have followed the GQM methodology because this methodology directly links engineering goals to evaluation criteria through the questions that are first used to characterize the goals and then to derive the criteria. Thus, if properly implemented, this methodology ensures that the evaluation model is complete with respect to the identified engineering goals.

Second, it is quite obvious that the criteria dimensions are not orthogonal but to some extent correlated, positively or negatively. A framework supporting full automation even for complex use cases requires a highly expressive language. On the other hand, less expressive languages will likely be easier to use and yield better runtime performance. Therefore, SWS frameworks need to aspire a balance between competing requirements. Superior performance with respect to one criteria will typically have implications on the relative performance with respect to the other

criteria. It is thus important to evaluate the dimensions identified above conjointly and as completely as possible to make the corresponding tradeoffs explicit.

However, the presented evaluation criteria model illustrates the broad variety of possible evaluation scopes and foci. One could, for instance, primarily be interested in evaluating the performance of composition algorithms, or one might be interested in the usability of frameworks supporting developers setting up data or process mediation. Even within a specific scope, one will have to make certain context assumptions, for instance, about the type and complexity of services, or the primary domain of interest. Alternatively, one could also make the investigation of the influence of such context assumptions the primary evaluation goal. In short, the variety of variables illustrates the variety of evaluation questions potentially of interest. Any such question ideally requires a specifically designed experiment or benchmark to be properly investigated and answered. Providing a complete set of such experimental designs and benchmarks is far beyond the scope of this thesis. This obviously conflicts with the above motivated desire for complete evaluations that cover the criteria dimensions conjointly. Therefore, a set of evaluation questions to be exemplarily covered by this thesis had to be selected. The selection will be motivated in Section 4.3.

4.2. Requirements for SWS Technology Evaluations

The model presented in Section 4.1 allows to classify evaluations according to their evaluation goals. This section discusses the requirements to the evaluation process itself and presents a requirements catalogue for evaluations. This catalogue will provide the framework to meta-evaluate the contributions of this thesis in Section **??**.

In order to obtain an objective and independent evaluation requirement framework, the presented requirements build directly on the Evaluation Standards edited by the German Evaluation Society (DeGEval)[2]. DeGEval was founded in 1997 and has more than 100 institutional and several hundred individual members. Members are recruited from renowned economic and social scientific research institutes, institutes of higher education, consulting and political consulting agencies, ministries, administration departments, and federal research institutes with all relevant disciplines and professions being represented. DeGEval aims to promote the information flow and dialogue around the topic of evaluation, to consolidate the multitude of different perspectives, experiences and expectations among different research areas on the matter and to professionalize evaluations in all areas. To this end, it has developed its evaluation standards:

"These standards are intended to assure and promote evaluation quality in all application areas of evaluation. They shall foster dialogue and provide a specific

[2] `http://www.degeval.de/`

frame of reference for discussing the quality of professional evaluations. They are also designed [...] for the evaluation of evaluations (meta-evaluation) and to make professional practice more transparent for a wider public" [Bey03].

Besides the standards by the DeGEval, there are other related standards, most notably the "Guiding Principles for Evaluators"[3] by the American Evaluation Association[4], the American counterpart of the DeGEval. These are similar in spirit to the DeGEval Standards, but comparatively high-level and less operational. Therefore, the DeGEval Standards have been chosen as basis for the requirement framework in this thesis.

The DeGEval standards are divided into four categories according to basic attributes that evaluations shall demonstrate: *utility*, *feasibility*, *propriety*, and *accuracy*. They are intended to cover evaluations in a wide area of domains and with different evaluation purposes.

This thesis is concerned with the evaluation of SWS technology. Furthermore, of the different main purposes of such evaluations, it is primarily interested in community driven evaluation campaigns with the purpose of enhancing scientific progress by supporting learning and reflection processes in the field of this technology (see Section 2.4.1). Figure 4.2 shows an overview of the DeGEval standards and their relationship to community driven evaluation campaigns.

In order to apply the DeGEval standards, they need to be concretized to the specific evaluation use case at hand: "They formulate key points which evaluator shall respect and goals they shall pursue. They are intended to provide a frame of reference for conducting and assessing evaluations. How they are implemented is a deciding factor. It cannot take place schematically. [...] The evaluation team and all participants have the job of finding an appropriate solution which takes account of the purposes and context of the evaluation in hand" [Bey03].

In the remainder of this section each DeGEval standard is briefly described. The italic rendered standard description is verbatim quote from the official standards [Bey03]. Each standard may be further illustrated by an additional comment (not quoted verbatim from the original standards unless explicitly marked). Each standard is operationalized in the spirit of the GQM by a list of questions that help assessing how the standard is applied. These questions are contributions and not part of the original standards. It is primarily these questions which adapt the general Evaluation Standards to the evaluations targeted by this thesis. In order to be as objective and unbiased as possible, the questions heavily leverage existing work in the area in particular by Sim et al. [SEH03, Sim03], García-Castro [GC07, GC08], Weiderman et al. [WHBK87], Avesani et al. [AGY05] and Euzenat and Shvaiko [ES07]. All these authors have compiled similar question or requirement lists for different

[3] `http://www.eval.org/Publications/GuidingPrinciples.asp`
[4] `http://www.eval.org/`

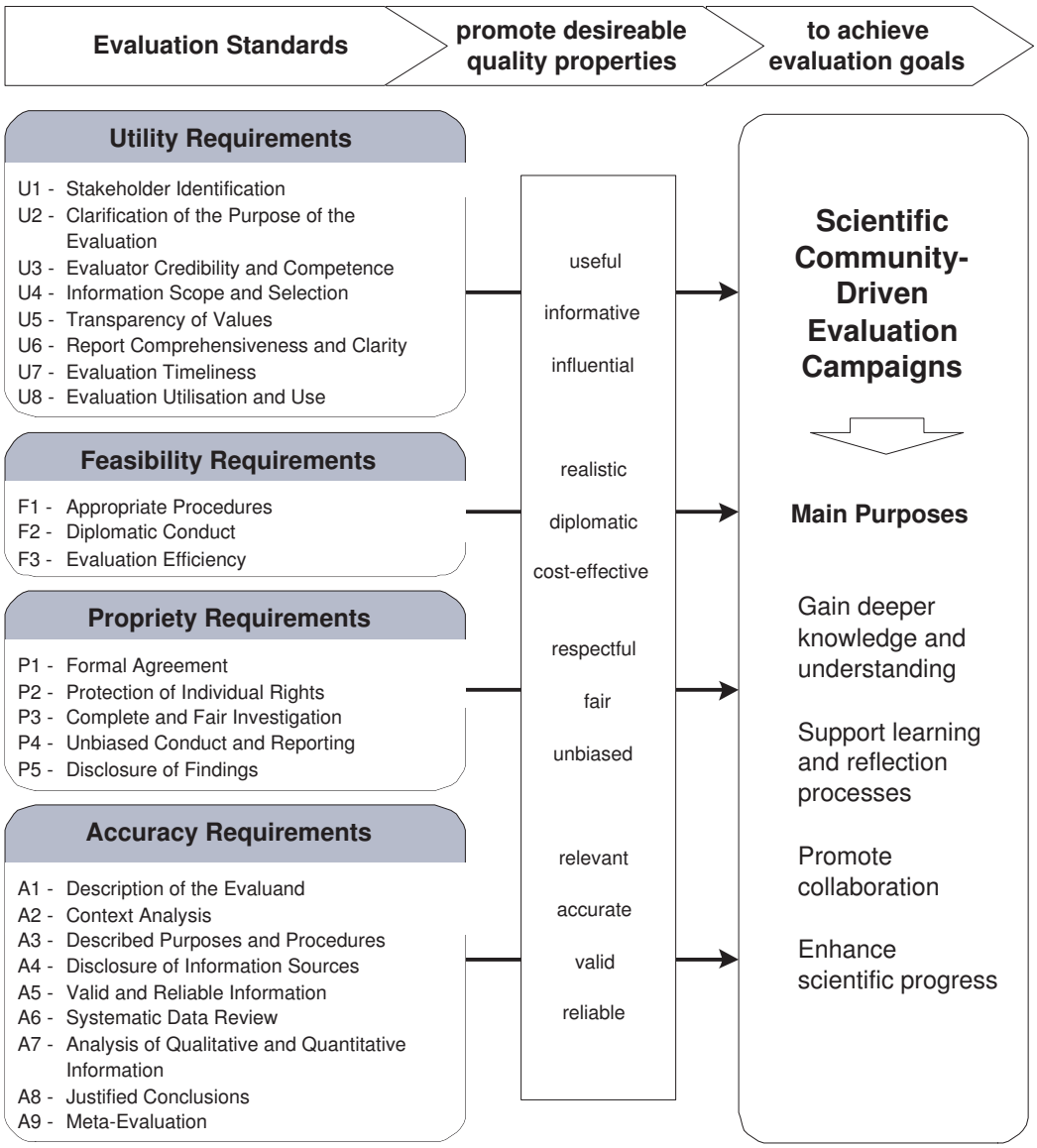

Figure 4.2.: Requirements to evaluations adopted from the Evaluation Standards of the German Evaluation Society

types of evaluations. Note that some of the DeGEval standards overlap. Questions that apply to several standards have not been duplicated but are listed only once under the standard to which they apply most.

4.2.1. Utility Requirements

The Utility Requirements are intended to ensure that an evaluation is guided by both the clarified purposes of the evaluation and the information needs of its intended users.

Utility 1 (Stakeholder Identification) *Persons or groups potentially interested in or affected by the evaluation should be identified and contacted, so that their interests can be clarified and taken into consideration when designing the evaluation.*

- Is there a description of the scope of the evaluation, potential participants in the evaluation campaign as well as the user of the evaluation results?
- Do the people in the community potentially interested in the evaluation have a chance to become involved in the development of the benchmark and the planning and execution of the evaluation campaign?
- Have institutions and individuals involved in the development of technology within the scope of the evaluation been contacted?

Utility 2 (Clarification of the Purposes of the Evaluation) *The purposes of the evaluation shall be stated clearly, so that the stakeholders can provide relevant comments on these purposes, and so that the evaluation team knows exactly what it is expected to do.*

Community evaluation campaigns should foster a cooperative atmosphere of collaboration and aim at providing constructive insights to the stakeholders how to improve and develop the participating technologies.

- Is there a definition of the goals and purposes of the evaluation?
- Is the expected usage context of the evaluation and its findings clearly described?
- Do evaluation results provide insights about the technology characteristics and development practices that lead to them?
- Do results provide technology improvement recommendations? Is improvement of the evaluated tools or techniques one of the goals of the evaluation?

Utility 3 (Evaluator Credibility and Competence) *The persons conducting an evaluation shall be trustworthy as well as methodologically and professionally competent, so that the evaluation findings achieve maximum credibility and acceptance.*

Any evaluation must be developed by experts who apply their knowledge of the domain and are able to identify the key problems, but a community evaluation campaign in particular must reflect the needs of the community as a whole. Thus, an evaluation should be assessed and agreed on by the whole community.

- Does the evaluation team properly reflect the diversity of the interested research community? Are the main lines of research properly represented?
- How many people / research groups are involved in the development of the evaluation and the planning and execution of the evaluation campaign?
- Do people in the wider community have a chance to become actively involved? Have people active in the corresponding research area that are not involved in the evaluation been approached to become involved?

Utility 4 (Information Scope and Selection) *The scope and selection of the collected information shall make it possible to answer relevant questions about the evaluand and, at the same time, consider the information needs of the client and other stakeholders.*

Ideally, the selection of information should be confirmed by community consensus. The evaluation should allow to incrementally discover the weaknesses and strengths of the tested technologies to provide clues to their evolution. It should cover the problem domain of interest comprehensively. It should be possible to complete most of the tasks and to produce a good solution but the task sample should be "hard" to solve for state of the art systems. A sample that is too difficult for all or most technologies yields little data to support comparison. A sample that is too easy will not allow discriminating between technologies. A sample that is achievable, but not trivial, provides an opportunity for systems to show their capabilities and their shortcomings. It is useful to include some tasks which can not be solved with state of the art systems yet. Such tasks mark challenging long term goals of the field of science under investigation.

- Was the design of the evaluation discussed at a meeting that was open to the community; more than once?
- Are there other competing evaluation approaches which have been used by different groups? How do they differ and why have they not been used?

- Is the scope of the evaluation clearly defined?
- Are the problems addressed by the evaluation valid representatives of problems found in actual practice?
- Are the evaluation tasks and the used input data good representatives of tasks and data that the evaluated technologies are reasonably expected to handle in a natural setting?
- Is the selection of tasks or input data justified and supported by empirical work or a model or theory?
- Does the evaluation provide tasks or inputs at different complexity or size?
- Can the tasks and problems of the evaluation be scaled to be more or less complex, more or less numerous, shorter or longer?
- Can the tasks be solved? Does an exemplary solution exist?
- Is a good solution possible? Is a poor solution possible?

Utility 5 (Transparency of Values) *The perspectives and assumptions of the stakeholders that serve as a basis for the evaluation and the interpretation of the evaluation findings shall be described in a way that clarifies their underlying values.*

Underlying values in the context of technology evaluation primarily concern the assumptions about the intended purposes and usage context of the technology.

- Are the assumptions behind the evaluation use case scenarios been described and made explicit?
- Have the design goals of the evaluated technologies been described and compared in detail?

Utility 6 (Report Comprehensiveness and Clarity) *Evaluation reports shall provide all relevant information and be easily comprehensible.*

Successful communication of evaluation findings demands comprehensiveness and clarity in the written reports or other forms of feedback. Important terms shall be unambiguously defined and used consistently, so that addressees are able to understand the language. The evaluation documentation shall be comprehensive, self-contained and easily understandable.

- Is the setup of the evaluation including prerequisites, assumptions, input data, roles and tasks, information collection and data analysis clearly described?

- Are the findings of the evaluation completely comprehensible for all stakeholders and also for interested outsiders?

Utility 7 (Evaluation Timeliness) *The evaluation shall be initiated and completed in a timely fashion so that its findings can inform pending decision and improvement processes.*

Evaluations need to be scheduled pertinent to their purposes and the intended usage of their findings. With respect to the previously discussed goals of community evaluation campaigns, these should not be a one-shot exercise but a continuous evolving effort in order to identify the progress made by the field. Reflecting the evaluation needs of a community, evaluations have to be continuously adapted to new findings and advancements of the field. The evolution of the evaluation will mirror the progress of the field. An evaluation campaign should eventually stop only if no more progress is made anymore. The procedural setup of an evaluation campaign should accommodate this co-evolution of the evaluation and the evaluated field of science.

- Is the evaluation intended to be used once or frequently?
- Is there a continuous series of subsequent evaluation events?
- Does the scheduling of the evaluation events accommodate the constraints of the expected participants? Is the event scheduling result of a consensus based planning process?
- Is there a process in place to collect feedback and make refinements to the evaluation based on that feedback?
- How many times and how frequently has the evaluation been updated?

Utility 8 (Evaluation Utilization and Use) *The evaluation shall be planned, conducted, and reported in ways that encourage attentive follow-through by stakeholders and utilization of the evaluation findings.*

In order to achieve this, the evaluation team should actively pursue to build a community around the evaluation, including providing infrastructure like web sites, mailing lists or discussion forums and arranging workshops or symposia to discuss results, exchange ideas and provide feedback. If people do not know about an evaluation campaign, it is merely a private project. Public calls for participation and word-of-mouth contact is key to raising the visibility of the project.

- Was the project well publicized?

- Has it been presented at public well-known events? How many times?
- Is there a web page about the evaluation with comprehensive information about it? Has the web page been publicized?
- Was participation in the evaluation open to all interested parties?
- Can the evaluation be executed offline or is it part of an evaluation event? How frequently are evaluation events or campaigns repeated?
- If applicable, is there a fee or other restrictions (e.g., a license agreement) for participating in the evaluation event or executing the evaluation?
- If applicable, is everything needed to execute the evaluation offline available? Is software and infrastructure supporting the evaluation available? Is software source code available? Is all input data and the necessary documentation available?
- Are the results of the evaluation well publicized? Were the findings discussed at a meeting that was open to the community; more than once?
- Are the evaluation results accessible in detail? Is it possible to examine the raw data in addition to the consolidated evaluation results?
- Is there a fee or other restrictions (e.g., a license agreement) for accessing the evaluation results, the solutions or the raw data?

4.2.2. Feasibility Requirements

The Feasibility Requirements are intended to ensure than an evaluation is planned and conducted in a realistic, thoughtful, diplomatic and cost-effective manner.

Feasibility 1 (Appropriate Procedures) *Evaluation procedures, including information collection procedures, shall be chosen so that the burden and cost placed on the stakeholders is appropriate in comparison to the expected benefits of the evaluation.*

Evaluation processes shall meet scientific merit criteria, but the most relevant methods from a scientific point of view are often unsuitable because they are too time-consuming or costly. Usually there is a tradeoff between higher validity or reliability and higher cost. The evaluation shall therefore clarify advantages and disadvantages and justify the relevance of the chosen procedure.

- Are the pros and cons of the chosen evaluation setup discussed?

- Are alternative evaluation procedures that involve higher or lower effort or monetary costs by the participants or the evaluation team discussed?
- Are tools available that support the evaluation?
- Is the execution of the evaluation automated where possible?
- Is the analysis of the collected data automated where possible?

Feasibility 2 (Diplomatic Conduct) *The evaluation shall be planned and conducted so that it achieves maximal acceptance by the different stakeholders with regard to the evaluation process and findings.*

Ensuring balanced consideration of all stakeholder interests when implementing the evaluation fosters acceptance, approval, and cooperation among the various parties. The evaluation team shall proceed diplomatically and aim at providing constructive insights how to improve and develop the participating technologies. Performing evaluations in a cooperative rather than competitive atmosphere greatly strengthens the positive social effects of community evaluation campaigns.

- Does the setup of the evaluation campaign encourage participants and other stakeholders to cooperate and mutually learn from each other?
- Is there an awarded winner of the evaluation event?
- Is the presentation of evaluation results discussed with the participants before they are made public?

Feasibility 3 (Evaluation Efficiency) *The relation between cost and benefit of the evaluation shall be appropriate.*

The setup of the evaluation should include a cost-benefit assessment and present a clear estimate of the predicted time and cost as well as advantages involved with participation. To increase the willingness of people to participate in an evaluation campaign, it may be useful to subdivide the evaluation in parts of which some are optional. This reduces the entry cost of the evaluation campaign without compromising the reliability and validity for those participants that are able and willing to allocate more resources.

- Do the participants in the evaluation need special skills, knowledge, and training?
- What is the minimal and optimal time (person hours) required from the participants and the evaluation team to execute the evaluation?

- What is the total monetary cost involved for the participants and the evaluation team to implement the evaluation?
- Can the evaluation be scaled to be more or less complex and costly?

4.2.3. Propriety Requirements

The Propriety Requirements are intended to ensure that in the course of the evaluation all stakeholders are treated with respect and fairness, that the evaluation achieves maximum objectivity and provides an unbiased and appropriate analysis of the technologies under examination.

Propriety 1 (Formal Agreement) *Obligations of the formal parties to an evaluation (what is to be done, how, by whom, when) shall be agreed to in writing, so that these parties are obligated to adhere to all conditions of the agreement or to renegotiate it.*

- Has a written agreement about the responsibilities and commitments of everyone involved in the evaluation been prepared?
- Are the responsibilities and commitments of everyone public and transparent to the community?

Propriety 2 (Protection of Individual Rights) *The evaluation shall be designed and conducted in a way that protects the welfare, dignity and rights of all stakeholders.*

"Evaluators shall ensure that they do not encroach on the dignity and self-respect of the people with whom they interact in the course of the evaluation" [Bey03].

- Is there a process for users to vet their results before they are released?
- Is there an option for users to veto the release of their results?

Propriety 3 (Complete and Fair Investigation) *The evaluation shall undertake a complete and fair examination and description of strengths and weaknesses of the evaluand so that strengths can be built upon and problem areas addressed.*

A technology's strengths and weaknesses are often closely related. Existing strengths can counteract weaknesses and correcting weaknesses can sometimes undermine existing strengths. Thus, an evaluation shall cover the complete problem domain and provide an as comprehensive as possible assessment of the technology under evaluation.

- Does the evaluation result in a single score or does it provide detailed information about the performance of the tools or methods under investigation?
- Does it make both the strengths and weaknesses of the evaluated technologies explicit and discusses them?
- Does it allow a detailed, meaningful comparison of the evaluated technologies?
- Which of the potentially interesting characteristics of the technologies under evaluation does the evaluation cover and which does it not cover? Is the selection justified?

Propriety 4 (Unbiased Conduct and Reporting) *The evaluation shall take into account the different views of the stakeholders concerning the evaluand and the evaluation findings. Like the entire evaluation process, the evaluation report shall evidence the impartial position of the evaluation team. Value judgments shall be made as unemotionally as possible.*

- Is the evaluation independent (in particular not reverse engineered) from particular solutions?
- Is the evaluation tied to a particular platform or technology?
- Are the evaluation tasks and the associated data specified at a level of abstraction that ensures its applicability to different technologies without being biased towards specific ones?
- For which technologies in the field under investigation can the evaluation potentially be used? For which can it not be used and why not?
- Can the evaluation be meaningfully applied to research prototypes as well as mature products?
- Does the evaluation provide tasks or inputs at different complexity or size?
- Does the evaluation report discuss potential biases or aspects that may be conceived as such by a stakeholder?
- Is there a reviewing process for the evaluation results and report in place?

Propriety 5 (Disclosure of Findings) *As far as possible, all stakeholders shall have access to the evaluation findings.*

Furthermore, all necessary information to be able to fully understand and ideally reproduce the findings should be available. The terms of publication of evaluation findings, how, when, by whom, according to which criteria and with respect to which limitations and restrictions shall be agreed at the beginning of an evaluation and documented in written form. The availability and transparency of the evaluation findings is key to the acceptance and thus impact of the evaluation. If the evaluation is not sufficiently transparent, its results will be questioned and it could be interpreted incorrectly. Any stakeholder must be able to understand how the evaluation works and in particular how its results are obtained.

- Are all scores and evaluation results clearly documented?
- Is all data available that stakeholders need to comprehend and reproduce the evaluation findings?
- Have the evaluation findings been disclosed according to the previously agreed upon terms?

4.2.4. Accuracy Requirements

The accuracy requirements are intended to ensure that an evaluation produces and discloses valid, accurate, precise, reliable and useful information and findings pertaining to the evaluation purposes and questions.

Accuracy 1 (Description of the Evaluand) *The evaluand shall be described and documented clearly and accurately so that it can be unequivocally identified.*

- Does the evaluation clearly show what is under investigation and make any prerequisites for applying the evaluation to a technology explicit?
- Are the quality criteria being assessed in the evaluation clearly defined?
- Are the tools or techniques that are intended to be evaluated defined at the outset?
- Are the evaluation requirements and assumptions clearly specified?

Accuracy 2 (Context Analysis) *The context of the technologies being evaluated shall be examined and analyzed in sufficient detail.*

This particularly concerns any assumptions (often implicit) related to the intended usage of a technology under evaluation.

- Are the contexts of the evaluated technologies (engineering goals, expected usages, development status, ...) described in the evaluation report and their potential influence on the evaluation findings discussed?

Accuracy 3 (Described Purposes and Procedures) *Object, purposes, methodology and procedures of an evaluation, including the applied methods, shall be accurately documented and described so that they can be identified and assessed.*

Documentation of the process incorporates a detailed description of the complete setup of the evaluation including any prerequisites, the context of the evaluation environment, all data being used or collected during the evaluation, any analysis and interpretation performed on that data, and finally, the evaluation reporting.

- Are the measured characteristics of the systems or methods under evaluation clearly defined?
- Are the evaluation criteria clearly defined?
- Are the tasks to perform and the input data well defined and documented?
- Is the procedure how the evaluation is executed, how the resulting scores are compiled and how these are to be interpreted clearly defined and well documented?
- Is any infrastructure and software supporting the data collection and analysis well documented?

Accuracy 4 (Disclosure of Information Sources) *The information sources used in the course of the evaluation shall be documented in appropriate detail so that the reliability and adequacy of the information can be assessed.*

"Clear description of the data sources used allows addressees to form their own opinion on their quality" [Bey03].

- Is the raw data on which the evaluation findings are based accessible?
- Is it clearly traceable how the raw data was obtained?

Accuracy 5 (Valid and Reliable Information) *The data collection procedures shall be chosen and developed and then applied in a way that ensures the reliability and validity of the data with regard to answering the evaluation questions. The technical criteria shall be based on the standards of quantitative and qualitative social research.*

The data and data measurements must clearly and correctly reflect the criteria under examination. The data collected must be good indicators of the performance of the tested technology with respect to the evaluation criteria (*relevance*). The measurement results shall be independent from the person performing the measurement (*objectivity*). The measures shall produce consistent, reproducible and reliable information and be as resistant as possible to random errors and other factors not relevant to the study (*reliability* and *robustness*). They shall actually record the characteristics and behavioral patterns they claim to measure (*validity*).

- Are the assumptions made by the evaluation (e.g., regarding the expected user, the usage context, etc.) realistic?
- Is the selection of performance measures justified and supported by empirical work or a model or theory?
- Would one person applying the evaluation on the same technology twice get the same results?
- Would different people applying the evaluation on the same technology get the same results?
- Are results affected by unpredictable environment behaviors?
- Are threats to the evaluation's validity (factors which may influence the evaluation results but are not intended to be measured) clearly identified and discussed?
- Which of the results are affected by the quality of tool support and maturity of implementation of a particular technique under investigation; to what extent?
- Which of the results may be affected by optimizing a technology for the measures used in the evaluation?
- Are there procedures in place for auditing evaluated technologies to prevent against cheating and to identify solutions that are overly optimized towards the procedure and measures of the evaluation?

Accuracy 6 (Systematic Data Review) *The data collected, analyzed and presented in the course of the evaluation shall be systematically examined for possible errors.*

Any infrastructure supporting the evaluation, in particular tools to collect or analyze the data shall be well tested, free of bugs and run in the expected way. Collecting, processing, assessing and interpreting information and presenting findings creates a

wealth of potential pitfalls. These may be methodological errors or simply a lack of care. It is therefore crucial to design the evaluation process so that potential pitfalls can be identified at an early stage and errors, which otherwise could invalidate the whole evaluation, be avoided or corrected as far as possible.

- Has the infrastructure and software supporting the evaluation been tested to run in the expected way? Are the test procedures and results documented?
- Is the infrastructure and, if applicable, its source code available to be additionally tested and verified by interested stakeholders?
- Is there a review process in place which critically reviews all data being assembled during the course of an evaluation?

Accuracy 7 (Analysis of Qualitative and Quantitative Information)
Qualitative and quantitative information shall be analyzed in an appropriate, systematic way so that the evaluation questions can be effectively answered.

The scores produced by the evaluation must clearly and correctly reflect the criteria under examination and be good indicators of the fitness for purpose between the tested technology and the performed tasks. "The data analysis process sorts, summarizes and assesses the quantitative and qualitative data gathered during evaluations. This forms the basis for interpretations and conclusions in the process of answering the evaluation questions. Selection of appropriate survey and analysis procedures shall be based on the evaluation questions, the current level of information on the evaluand and context variables in the evaluation field. Evaluator preferences shall play no role in this decision. [...] Choice and application of procedures shall be transparent and comprehensible so that selection decisions and findings can undergo critical appraisal." [Bey03].

- Are benchmarks and formulae explained in a way that everyone can understand?
- Are values and limitations of the methods used stated explicitly?
- Are the compiled measures used in the evaluation good indicators of the performance of the technology with respect to the quality criteria of interest (fitness for purpose)?
- Is it possible for a tool or technique that does not have fitness for purpose to obtain a good performance score?
- Is it possible for a tool or technique that does have fitness for purpose to obtain a bad performance score?

- Does a score represent the capabilities of a single technology fairly and accurately?
- Can the scores be used to directly compare two technologies?

Accuracy 8 (Justified Conclusions) *The conclusions reached in the evaluation shall be explicitly justified so that the audiences can assess them.*

Discussing alternative interpretations and justifying the conclusions drawn in the evaluation reinforces the conclusions' credibility.

- Are evaluation conclusions substantiated and presented clearly with the fundamental suppositions and the procedures applied?
- Is the scope of the conclusions defined and discussed?
- Are alternative interpretations discussed, including the reasons why these were rejected?

Accuracy 9 (Meta-Evaluation) *The evaluation shall be documented and archived appropriately so that a meta-evaluation can be undertaken.*

This fosters scientific progress and knowledge accumulation in the relevant field and supports the continuous evolution and iterative improvement of the evaluation.

- Have the key purposes, steps, methods, data, and findings of the evaluation been archived?
- Have meta-evaluations and comparisons with previous or alternative evaluations been performed?

4.3. Analysis of SWS Evaluation Approaches by Evaluation Criteria

The evaluation criteria model presented in Section 4.1 allows classifying and analyzing the various existing approaches to SWS evaluations systematically by discussing how the various criteria have been approached. This will be done in the following. The discussion includes for each criteria a status report, options for improvements and conclusions. Section 4.4 will complement this discussion by a meta-evaluation of the community evaluation initiatives with respect to the evaluation requirements presented in Section 4.2. Based upon both discussion, the selection of the concrete improvements covered by this thesis will be motivated in Section 4.5.

4.3.1. Performance / Scalability

This criteria regards the runtime performance and scalability characteristics of a framework. It is typically measured by the computing resources required (e.g. processor time or memory).

Status

A comparative evaluation of the runtime performance of different matchmaking algorithms is provided by the S3 Contest. The experimental task to perform is to compare a given set of semantic request descriptions with a given set of semantic offer descriptions and identify the set of relevant services for each request. This task is executed by the participating matchmaker implementations multiple times and the average query response time for single queries as well as the average total time to match all requests is measured. In the 2007 edition, the results for two matchmakers were roughly similar (11 respectively 9 minutes) whereas a third matchmaker required more than 20 hours to perform the task on a significantly downsized version of the test data. Similar, even though less drastic variance in results was observed in the 2008 and 2009 edition of the Contest.

Unfortunately, a detailed interpretation of the results has not been provided. An analysis of the causes for the relatively weaker performance of the third matchmaker would be important to investigate whether that poor performance is inherent to the particular matchmaking algorithm or has to be attributed to an unoptimized proof-of-concept implementation of the algorithm. It is still hoped that participants of the contest are investigating the causes for encountered performance issues and will report on corresponding improvements in subsequent editions of the contest. Such analysis should probably be solicited more explicitly. It is worth noting that the S3 Contest evaluates the runtime performance and the correctness of the returned results (see Section 4.3.3), thereby allowing to put the runtime performance measures in relation to the achieved correctness.

Apart from the S3 Contest, the WS Challenge offers an evaluation of runtime performance of service composition algorithms (previously also service discovery algorithms). The task setup is similar to the one of the S3 Contest except that the WS Challenge, as discussed in Section 3.1.3, uses much lower level semantics. Furthermore, the WS Challenge uses a test data generator whereas the S3 Contest relies on handcrafted data. This approach has also been followed in some project based evaluations. Notably, the setup of the WS Challenge has been changed in 2009. While previous editions measured the time necessary to solve the discovery or composition challenge, the current approach is to provide a fix time slot (300 seconds) and measure the best solution computed within this time.

Discussion and options for improvements

It is obvious that runtime performance measures are highly dependent on the test data used. Unfortunately, no standard test collection for the evaluation of SWS exists yet. The approach of the WS Challenge to synthetically generate test data allows the creation of arbitrarily large data sets and is thus especially suited to investigate scalability issues. However, with generated data, it is not certain that the characteristics of that data reflect real world conditions. Therefore, the S3 Contest has chosen another approach.

To make experimental performance evaluations possible, OWLS-TC[5] and SAWSDL-TC[6], the test collections used by the S3 Contest in 2007, 2008 and 2009 and the only sizeable ones currently available, have been developed. So far, this effort was carried out primarily by a single group. This is not feasible in the long run. Due to the tremendous effort involved and in order to reflect different views and different perspectives, standard test collections can only be built by the community as a whole.

Similarly, due to the at that time complete lack of any noteworthy SWS test data in a formalism other than OWL-S [KKR08b, KZ08] the 2007 edition of the S3 Contest had to be limited to OWL-S matchmakers. In 2008 an initial port of OWLS-TC to the SAWSDL formalism was developed, thus, the scope could be extended to SAWSDL matchmakers in the 2008 edition of the contest. However, much more test data is necessary in the future. First, test data in the WSML/WSMO formalism is not available in sufficient quantity. Therefore, this important branch of SWS research and development can not be covered by the S3 Contest yet. Second, even within a formalism, a larger variety of descriptions is necessary. The current 2.2 version of OWLS-TC, for instance, does not contain service descriptions with formalized preconditions and effects but is limited to formalizations of the inputs and outputs of the services. Performance measurements obtained using this test collection do not provide sufficiently well-founded insights about the performance of the same matchmakers when processing more complex service descriptions. For the future, large, varied and truly comparable standard test collections of the same set of services in different formalisms need to be developed.

Scalability has not been explicitly evaluated by the S3 Contest so far. However, this could be done with limited additional effort. It requires splitting the test collections in sub collections of different sizes and exploring the degradation of the runtime performance with increasing size of the test data. Obviously the remarks about sensitivity towards the composition of the employed test collections apply in the same way as discussed above.

[5] http://projects.semwebcentral.org/projects/owls-tc/
[6] http://projects.semwebcentral.org/projects/sawsdl-tc/

Conclusions

Performance and scalability measures and their associated potential pitfalls are very well understood and have been used in all areas of software engineering for decades. Their application in the area of SWS is primarily hampered by practical issues. In the area of SWS matchmaking, for instance, despite of a wealth of work only few implementations of the proposed matchmaking algorithms are readily available. This is a blocker for better evaluations also with respect to other criteria. Additionally, the lack of standard test collections of SWS has proven to be difficult to overcome. Furthermore, the effects of the properties and composition of the test collections on the evaluation results need to be studied carefully. This requires different collections with different properties but will ultimately allow building standard test collections that are diverse and balanced, ensuring reliable evaluations.

4.3.2. Usability / Effort

This criteria regards the usability of the framework in terms of how much effort is required to set it up, maintain it and use it. This criteria is influenced by the complexity of a framework but also the available tool support.

Status

An initial attempt to evaluate the usability of a SWS framework has been made within the DIANE Benchmark. The approach is based on evaluating the initial effort to create the necessary ontologies and the continuous effort to update and maintain them. The initial effort is evaluated by measuring the time it takes an experienced developer to formalize an ontology given as a UML model in the language of the target framework. The continuous effort to maintain a framework is estimated by the DIANE Benchmark via a questionnaire that tries to assess the quality of the available tool support and documentation. Besides the approach of the DIANE Benchmark, significant effort has been devoted to develop a methodology to assess the flexibility of solutions within the SWS Challenge. The approach is based on evaluating the effort necessary to adapt a solution for a given problem scenario to variations of that base scenario. Notably, approaches based on SWS as well as more traditional software engineering technologies participate in the SWS Challenge. This allows to investigate not only the relative advantage of one SWS approach over another, but also to compare them with traditional technologies. A detailed description of the methodology employed by the SWS Challenge and the difficulties encountered is available as a W3C Incubator Group Report [PKMS08].

Discussion and options for improvement

While the SWS Challenge relies on complete natural language descriptions of scenarios, the DIANE Benchmark follows a much more restricted approach. It is thus easier to implement and involves less effort for participants. However, the task of formalizing an ontology given as a UML model prescribes the level of detail to be formalized. Lightweight frameworks, which do not exploit many details from the descriptions of services during the matchmaking, might be penalized with the effort of formalizing aspects which are of no use to them.

Generally, the choice of the right level of detail for a formalization of a problem still constitutes one of the core research problems in the area and should not be dictated by the testbed for an evaluation. Though experience with natural language scenario descriptions within the SWS Challenge showed that these descriptions were ambiguous in several cases, such ambiguities were discovered by the participants and could subsequently be resolved. This way even scenarios described in natural language only become sufficiently well-defined over time.

It thus seems appropriate to combine both approaches, provide complete natural language descriptions of use cases (as the SWS Challenge does) and evaluate the time necessary to implement these with a framework (in the spirit of the DIANE Benchmark). This setup reflects the strengths and weaknesses of the frameworks more adequately. A lightweight framework, for instance, might benefit from a reduced modeling effort but later suffer from poorer measures regarding the correctness of the achieved results.

Notably, this approach has not been taken so far. Because of the amount of work involved in implementing such an approach, the SWS Challenge has resorted to evaluating the effort of implementing changes on top of existing solutions instead of evaluating the effort of creating the initial solutions in the first place. Furthermore, there were concerns that measuring the time needed to perform the necessary adaptations would lead to an unwanted competitive atmosphere and would be overly sensitive towards the personal performance of the programmer implementing the changes. Additionally, it was found that there is not yet an easy way to distinguish the effects of inherent properties of a technology from the influence of the quality of the tool support available within or for a particular framework.

As a consequence, it was tried to measure the amount of code that needs to be changed instead of the time needed to implement those changes. Unfortunately, this change-based approach proved to be impossible to implement objectively, in particular in cases where code is not written as textual instructions but by assembling processes graphically in a GUI. A satisfying solution to this issue has not yet been found.

Regarding the complementary questionnaire approach of the DIANE benchmark it is felt that a questionnaire is a good since lightweight starting point. However,

the current implementation has several problems: The answering scheme (*yes* – *partially* – *no*) is too coarse-grained, some answers cannot be verified objectively and the weighting of the single questions in the total result is not based on experimental evidence.

Conclusions

Overall, efforts regarding the evaluation of the usability and ultimately the increase in programmer productivity achieved through SWS frameworks are in their infancy. One of the problems currently hindering more extensive usability evaluations is the already mentioned lack of implementations and tools for the proposed algorithms. Generally, the fact that the majority of SWS related tools are currently developed as research prototypes in academic settings makes usability evaluations inherently difficult. Many usability problems will be caused rather by a lack of implementation maturity of the tool support than by inherent properties of the technology.

The lack of ready-to-use tools might also explain the fact that current evaluations have focused on usability on a technical level, e.g. investigated how long it takes to perform an update to an ontology. However, ontologies and their management are just a means and technology to achieve higher level goals. Therefore, such evaluation efforts need to be complemented by evaluations of the increase in productivity on a higher, more goal-oriented level. Such evaluations would also improve the comparability of SWS technology with traditional software engineering technologies, a crucial factor for the adoption of SWS by industry.

The attempts of the SWS Challenge to measure the flexibility of solutions are a promising step in this direction, but also illustrate that the question how to reliably and objectively measure an increase in productivity achieved by using different SWS approaches is a still unsolved research problem.

4.3.3. Correctness / Automation

This criteria regards the quality and extent of the support offered by the framework. Correctness relates to the degree to which a framework acts precisely like the user it acts on behalf of. Automation concerns the degree to which tasks previously performed by humans are now performed by the framework. A high degree of automation obviously requires a high level of correctness. Correctness is closely related to the often used notion of *expressivity* that captures how precisely and comprehensively a service's capabilities and a user's needs can be formalized in a framework.

Status

Prior to the establishment of the S3 Contest in 2007, there have not been comparative correctness evaluations of different SWS matchmaking approaches at all. To get started, the S3 Contest borrowed the well-established evaluation approach from the series of TREC conferences[7] in information retrieval (IR) using the previously discussed OWLS-TC. Correctness of service matchmaking is evaluated by means of the traditional IR measures precision and recall. Precision measures the proportion of retrieved services, which are indeed relevant, and recall measures the proportion of relevant services that are correctly retrieved. In 2007 and 2008 the contest relied on binary relevance judgments, i.e. service offers are judged as either relevant or irrelevant to a request, but no further ranking is considered. This corresponds to the state of the art.

The WS Challenge evaluates the correctness and completeness of computed service compositions (previous editions also included a similar track on service discovery). Unlike the S3 Contest, which establishes the reference gold standard by human judgments, the WS Challenge specifies unambiguous conditions on correct solutions based upon the inheritance relationship of IO types. This makes the computation of correct compositions conceptually easy. Correctness as evaluated by the WS Challenge therefore refers to the correctness of an implementation rather than the accuracy possible with a semantic approach like in case of the S3 Contest.

The SWS Challenge focuses on functional coverage of frameworks (see below) and currently does not aim at providing quantitative measures for the correctness achieved by participating approaches. An entry to the challenge is usually developed until it correctly solves a scenario and not submitted otherwise.

The DIANE Benchmark presents two approaches to evaluate correctness. The first is similar to the approach of the S3 Contest but focuses on whether correct results can be achieved in an explicitly decoupled setting. It will be covered in Section 4.3.4. The other approach complements the S3 Contest in that it focuses on how well the real world semantics of services can be captured in the formalism used by a framework. It therefore attempts to evaluate correctness by experimentally evaluating the expressivity of the employed formalism. To define the benchmark, a group of test subjects not familiar with semantic web technology were asked to formulate service requests for two different application domains. The queries the test subjects devised were formulated in natural language. This resulted in about 200 requests. Additionally, domain experts developed ontologies they deemed necessary to handle the two domains.

The evaluation approach of the benchmark is to measure the proportion of the 200 requests which can be formalized in a given framework correctly. Each request can be rated green (the request can be directly formalized), yellow (the request could be

[7] `http://trec.nist.gov/`

formalized with extensions to the domain ontologies) or red (the request cannot be appropriately expressed using the language constructs provided by the framework). These ratings are expected to capture how well a framework's formalism is able to describe realistic services of different types.

Discussion and options for improvement

The adoption of the well-established correctness measures precision and recall from IR is a self-evident first approach towards correctness measures in the field of SWS retrieval. Obviously, the general remarks about the sensitivity of evaluation results towards the composition of the employed test collection and the discussion about the lack of standard test collections across formalisms made in Section 4.3.1 apply here, too.

However, as argued in [KLKR07], traditional IR and SWS retrieval differ in that the former typically operates directly on the original resources, whereas the latter is based on formal semantics that are explicitly manually attached to the resources to support their precise and correct retrieval. Following the TREC evaluation approach the S3 Contest presets the semantic descriptions used for the retrieval. The major benefit of this approach is twofold: it mimics real world environments, where SWS descriptions are not formalized by the developers of a SWS matchmaker (see Section 4.3.4) and it limits the effort involved in participation in the Contest. It does have the drawback, however, that recall and precision alone in such a setting can only be of limited significance. The problem is that the question whether a semantic service description matches a semantic request description should be determinable unambiguously based on the formal semantics of the employed description formalism. In this aspect, it is unclear to what extent false results of the matchmaking (and thus a low precision and recall) should be attributed to inapt service and request formalizations or to shortcomings of the evaluated matchmaking algorithms.

Thus, an ideal evaluation of SWS retrieval correctness needs to cover two aspects: First, how well the real world semantics of services can be captured in the formalism used by a framework. Second, how effectively the framework's matchmaker can then exploit this information during the matchmaking. An evaluation where the descriptions are preset is by design restricted to evaluating only the second aspect. On an implementation level, diverse test collections that contain service descriptions at various levels of detail and with varying complexity are required to evaluate this aspect reliably. Such collections are only partially available.

With respect to the first aspect, i.e. how to experimentally measure the quality of the formalization of a service's semantics possible in a framework, the DIANE Benchmark that relies on natural language service descriptions constitutes an important first achievement. Despite of that, an analysis of the evaluation of DSD performed with the DIANE Benchmark sheds light on two problems in the current

setup of this part of the benchmark. First, the distinction between green and yellow ratings seems arbitrary in many cases. It remains unclear, why certain concepts were included in the initial ontologies (leading to green ratings) while others were not (leading to yellow ratings) and why this is a relevant measure for the expressivity of a framework. It seems more appropriate to evaluate the effort necessary to implement required extensions to the ontology and use this as a measure for the usability of a framework. The second problem is a lack of objectivity regarding green ratings. Green ratings are supported by providing formalizations of these requests in the target formalism. However, the judgment that these formalizations fully capture the semantics of the service (justifying a green rating) is made by the subjective estimate of the expert formalizing the requests. Ideally, such judgments should be supported experimentally by an additional recall/precision analysis.

Conclusions

Until recently, there have not been any comparative evaluations of the correctness achieved by a SWS framework at all. It is very promising that this important issue is starting to receive the attention it deserves. However, as can be seen from the discussion above a meaningful correctness evaluation is far from trivial and the above mentioned problems illustrate the need for further research in this direction: First, current evaluations have either focused on the correctness of the matchmaking, or the correctness (or expressivity) of the formalization, but not on both. It needs to be investigated how this can be improved to achieve more reliable and comprehensive results. Second, current evaluations of correctness via recall and precision rely on binary relevance judgments. This approach has been a natural starting point, but does not reflect that virtually all SWS matchmakers support multi-valued matchmaking degrees and does not allow evaluating the important aspect of the quality of the ranking performed by SWS matchmakers. Further research on better measures, e.g. based on graded instead of binary relevance, is necessary [TAH06, KKR08a]. Third, the previously mentioned lack of standard test collections of SWS is even more critical for correctness evaluations than for performance evaluations. Reliable and meaningful correctness evaluations require diverse and realistic test data. This test data needs to be available in natural language to experimentally evaluate the expressivity of a formalism employed by a framework. Additionally, complete and high quality semantic descriptions for a common set of services are required in different formalisms to effectively compare the correctness achieved by the various algorithms. Generally, the desirable properties of test collections need to be investigated more thoroughly and procedures how to obtain the necessary data and ensure its quality need to be developed [KKR08b].

4.3.4. Coupling

This criteria regards how tightly coupled the providers and the consumers of services are when using a framework, e.g., whether they have to agree on common ontologies.

Status

An evaluation of the coupling between service providers and requesters within a certain SWS approach was so far not in the scope of the SWS Challenge. Within the participating teams the same developers typically formalize all goal and offer descriptions. Similarly it has not been explicitly in the focus of the S3 Contest or the WS Challenge so far.

The DIANE Benchmark presents an experimental setup to evaluate the degradation of delivered correctness in an explicitly decoupled setting. A number of inexperienced users are given an introduction to a framework and description formalism to be used. Subsequently they are divided into two groups that are not allowed to communicate with each other. A number of natural language service descriptions is provided as test data to the groups. The first group is asked to formalize them as offer descriptions, the second as request descriptions. Afterwards, the framework is used to match the resulting offer and request descriptions and precision and recall of the matchmaking are determined using the obvious binary relevance[8].

Discussion and options for improvements

The experience from applying this experimental setup to DIANE/DSD highlights an important issue: in practice, even using predefined ontologies, a high correctness is not easy to achieve in a decoupled setting. In the experiment a service that books a train ticket has been formalized as a service after whose execution a ticket is *reserved* by the first group. In contrast, the second group formalized the same service as a service after whose execution a ticket is *owned*. Subsequently, these different formalizations of the same real world semantics resulted in a false fail when the two descriptions were matched. This emphasizes the negative effects of variance in possible ways to formalize the real world semantics of a service. Such variance will inevitably be encountered in real world environments. It can be assumed that formalisms differ with respect to the likelihood of such modeling differences and that frameworks differ in how well they are able to handle them. Thus, a corresponding evaluation provides important clues about the performance of a framework in real world settings.

On a practical level, the DIANE Benchmark experiment needs to be considered preliminary. First, the remarks about binary relevance judgments in the context

[8]Request and goal descriptions resulting from the same natural language service description are considered relevant to each other, all other pairings are assumed to be irrelevant to each other.

of SWS matchmaking made above apply here, too. Second, the test data defined by the DIANE Benchmark for this experiment (ten services) is currently much too small to support reliable results in practice. Further work is required to address both issues.

Even though an evaluation of the degree of coupling has not been explicitly in the focus of the S3 Contest so far, it could be integrated into its setting very well. Since service request and offer descriptions are provided by the contest organizers, it could be assured that these have been developed in a decoupled way, in fact, it can be assumed that this is the case with OWLS-TC 2.2 to a large extent already. The degree of coupling could be evaluated by using different test collections explicitly designed to allow tracing back differences in performance to certain properties of these collections. OWLS-TC 2.2 for instance uses around two dozen ontologies. Several concepts used in the service descriptions are defined multiple times in different ontologies without being semantically aligned. In contrast, SWS-TC 1.1[9], another OWL-S based service collection, is based on a single, unified ontology. The degree of dependency on common ontologies could thus be evaluated by comparing the differences in performance of a matchmaking approach when used with those two service collections.

Conclusions

The importance of evaluating the degree of coupling and its effects within SWS frameworks is illustrated by the experience from the preliminary experiment in the context of DIANE/DSD. Yet, this aspect has received much too little attention so far. Typically, research, development, and evaluation of a given SWS framework is performed within a single research team and thus in a tightly coupled setting. In contrast, the envisioned use cases for SWS target strongly decoupled settings. It is thus essential to investigate the issues which may result from this discrepancy and to research methodologies to evaluate the tolerance of SWS frameworks towards these.

4.3.5. Functional Scope

This criteria regards the functional scope of a framework, i.e., which of the tasks typically involved in using functionality offered as a service (discovery, negotiation, composition, data and process mediation etc.) are supported by the framework and to which extent.

[9] `http://projects.semwebcentral.org/projects/sws-tc/`

Status

It is the main evaluation goal of the SWS Challenge to evaluate the functional scope of participating SWS frameworks. Here, we report the status as of 2006 (see [VLZP06]) prior to the improvements partially provided as contributions by this thesis. These will be described in Chapter 6.

The original approach of the SWS Challenge was to define a set of related problem scenarios, each consisting of increasingly complex problem levels. Every problem level adds another functional challenge on top of the previous levels. As of 2006, there was one mediation and one discovery scenario. The mediation scenario covered the integration of a legacy customer system with a RosettaNet PIP3A4[10] based purchase order system which required the resolution of data and process mismatches in the interfaces of the involved services. The discovery scenario required the discovery of suitable shipping services for given concrete shipment requests. Various requests required to reason about the shipper's operation range, package dimension and weight constraints as well as pricing models.

Participating solutions are certified at the Challenge workshops with respect to whether they are able to solve a particular problem level correctly. A review of the code during the workshop ensures that frameworks actually solve the problems by reasoning about the formalized problem semantics and not hard-wiring the known correct solution.

One of the goals of the SWS challenge is to address dynamic changes and to demonstrate how changes can be facilitated in a more flexible way by semantically enabled technologies. Thus, the organizers introduced a second level of the mediation scenario with some changes in messages and processes on which flexibility of different solutions had to be demonstrated.

The evaluation approach consisted of assessing the success in transitioning from one problem level to another, including the known problem levels and the previously unannounced changed version of the original mediation scenario. The basic success level 0 requires to correctly invoke the web services forming the scenario testbed. The correctness of the interaction is measured by the legality of the message exchange. Subsequent success levels 1–3 were achieved if the transitioning from one problem level to the next higher one involved the change of code (level 1), the change of only declarative data (level 2) or no change at all (level 3).

An evaluation of the scope of frameworks is neither performed by the S3 Contest nor by the WS Challenge nor within the scope of the DIANE Benchmark. The S3 Contest is limited to static service matchmaking, i.e. discovery which identifies relevant services based on static descriptions only. Similarly, the WS Challenge is limited to static composition of services. The DIANE Benchmark assumes support

[10] `http://www.rosettanet.org`

for dynamic discovery, ranking, selection and invocation and does not provide a fine-grained evaluation of frameworks, which only support some of these tasks.

Discussion and options for improvements

In many ways the setup of the SWS Challenge is similar to the use case based evaluations performed in many projects in the area. However, the design of scenarios explicitly with the aim of becoming standard benchmarks based upon which different approaches are compared is an important advancement. Nevertheless, the original setup of the SWS Challenge proved problematic in some aspects. The approach of building the more advanced scenarios on top of the existing ones requires building a solution starting with the basic scenarios and advancing to the more complex ones. However, different approaches face difficulties with different problem aspects. In fact, it turned out that some approaches were unable to solve the more complex problem levels just because they were unable to solve a seemingly much less complex underlying basic problem level. With this respect, evaluation results were difficult to interpret and sometimes misleading. More flexibility regarding which problem levels are addressed in which order by participating approaches is clearly desirable here.

Furthermore, the originally envisioned evaluation levels proved infeasible due to several reasons. First of all, it turned out to be impossible to objectively distinguish declarative data from executable code. This is particularly true for approaches where applications are assembled graphically on a GUI and later interpreted by some engine. Another problem was that participants could engineer their solution in a way that minimized the necessary changes when moving from one known problem level to another one. This could only be prevented by keeping problem levels secret and using code freezes of previous solutions. Generally, this resulted in a large overhead which was estimated disproportional to the perceived benefit.

Conclusions

The general approach of having common problem scenarios layered in different problem levels focusing on different problem aspects proved very useful, despite of some practical problems as discussed above. The specification of evaluation problems independently from the solution approaches significantly improves the relevance of the evaluation results compared to the typical project validations where demonstration and evaluation use cases and the technology to solve them are developed in tight interplay.

On a practical level, allowing more flexibility in how solutions are built and finding alternative evaluation criteria is necessary. Furthermore, the SWS Challenge original scenarios can only be considered a starting point covering only parts of

the problem space. Many more scenarios are needed to provide a more complete coverage of the problem space. More scenarios would also bring in the different perspectives and assumptions of different research groups in the area and thus help to confirm or revise the existing evaluation results. Fundamentally, also more research on methodologies that help ensuring the relevance and a certain completeness and balance of the testbed of scenarios is required.

4.3.6. Summary of Analysis

Table 4.1 presents an overview of which evaluation criteria are within the scope of which of the evaluation approaches discussed above. It also briefly recalls how the covered criteria (colored in gray) are evaluated. We conclude the analysis of the approaches with a summary of our finding for each evaluation criteria.

Performance / Scalability: Performance and scalability measures are very well understood but their application in the area of SWS is primarily hampered by practical issues. On the one hand, standard test collections are lacking, on the other hand not all proposed algorithms are implemented and readily available.

Usability / Effort: Usability evaluations are in their infancy. Many implementations in the area are prototypes for which sufficient tools support is not available. Furthermore, distinguishing the effects of prototypical implementations and insufficient tool support from the inherent complexity of an approach is extremely difficult. Both issues make meaningful evaluations of usability and increase in programmer productivity very challenging.

Correctness / Automation: Initial service matchmaking correctness evaluations have been performed using an evaluation setup from Information Retrieval. However, the limited applicability of this setup to *semantic* service retrieval and resulting problems are not discussed sufficiently. In particular assessments of the correctness of a semantic formalization and the processing performed on it need to be combined. Furthermore, the complete lack of test data for some formalisms is a critical problem for this evaluation criteria, too.

Coupling: The degree of coupling between service providers and requesters has not received much attention so far. Research, development and evaluation of SWS frameworks are largely performed within a single research group and thus in a tightly coupled setting. This contrasts the envisioned usage of SWS and should be overcome in future evaluations.

	S3 Contest	SWS-Challenge	WS Challenge	DIANE Benchmark
Performance Scalability	runtime for matchmaking	n/a	runtime for composition	not generalizable
Usability Effort	n/a	effort for adapting to problem changes	n/a	preliminarily assessed via questionnaire
Correctness Automation	retrieval correctness for specific formalisms	solutions are developed until correct	tests for correctness of algorithm but not semantic accuracy	correct representation of given sample requests
Coupling	decoupled setting, but not explicitly covered	n/a (offers and goals formalized together)	n/a (well-defined common schemata)	correctness of results in a decoupled setting
Functional Scope	n/a (limited to static discovery)	hierarchy of problem scenarios	n/a (limited to static composition)	n/a (limited to automated invocation)

Table 4.1.: Overview of evaluation criteria within the scope of existing approaches

Functional Scope: A problem scenario based approach for certifying the capabilities of approaches proved feasible in principle. However, there are several practical issues. First, the design of problem scenarios where each problem level builds upon the previous turned out to be problematic. Second, the concrete measures to evaluate success levels were found infeasible. Third, the relationship between problem scenarios and capabilities tested by them needs to made more explicit. On a more general level, methodologies to ensure the relevance, completeness and balance of the testbed scenarios are required.

4.4. Analysis of SWS Evaluation Initiatives by Evaluation Requirements

Having analyzed how different SWS evaluation criteria are addressed by current approaches, we perform a meta-evaluation of the state of the art with respect to the requirements on evaluations presented in Section 4.2. For the various evaluations performed primarily in privacy as part of project evaluations and reviews

this is largely infeasible due to a lack of published information about these evaluations. Furthermore, the purposes of such evaluations do not necessarily coincide with those of community-driven public evaluation campaigns. The analysis of existing evaluation approaches with respect to the requirements on evaluations will thus be restricted to cover the related community evaluation initiatives, i.e. the SWS Challenge[11], the S3 Contest[12] and the WS-Challenge[13]. Please note that the DIANE Benchmark was also not included since it has never been applied outside of the project it was developed within and it thus not directly comparable to the other three community based initiatives.

The aim of the presented assessment is not to criticize the reviewed initiatives but to identify current weaknesses to pave the way for their improvement. Furthermore, the assessment serves as an illustration that high quality standards for evaluations are not easy to achieve but require thorough investigations and continuous research and development.

Table 4.2 shows an overview of the assessment results. A checkmark "✓" denotes that the requirement is fulfilled. A checkmark in parenthesis "(✓)" denotes that the requirement is partially fulfilled and that the benchmark should be further improved with respect to this requirement. A minus sign "–" denotes that the requirement is not sufficiently fulfilled and improvement is required. The table illustrates that the requirements represent very critical high quality standards that mark long term goals and are not at all easy to achieve. All initiatives fall short of at least a few standards. To some extent this also illustrates that there are tradeoffs among competing requirements, like between the reliability of an evaluation and the effort required to perform it.

For improved readability, a complete discussion of all requirements is omitted here, but available in Appendix A. Here, only some of the key limitations of each initiative will be briefly discussed. Please note that the assessment represents the state as of end of 2009. Since the discovery track of the SWS Challenge was partially developed as part of this thesis work we will focus the coverage of this initiative on the complementary mediation track. The discovery track will be analyzed with respect to the requirements as part of the validation of this thesis in Section 8.6. Similarly, coverage of the S3 Contest will be limited to the OWL-S and SAWSDL matchmaking tracks. The complementary third track added in 2009 was developed as part of this thesis work and will be assessed in Section 8.7. Please also note that the assessment represents the best of our knowledge which may be inaccurate in particular with respect to the WS Challenge. Unlike with the SWS Challenge and the S3 Contest, the candidate is not involved in the organization of this initiative.

[11] `http://sws-challenge.org`
[12] `http://dfki.de/~klusch/s3/`
[13] `http://ws-challenge.georgetown.edu/wsc09/`

Evaluation Standard	**SWS**	**S3**	**WSC**
Utility 1 (Stakeholder Identification)	✓	✓	✓
Utility 2 (Clarification of the Purposes of the Evaluation)	✓	✓	(✓)
Utility 3 (Evaluator Credibility and Competence)	✓	✓	✓
Utility 4 (Information Scope and Selection)	(✓)	(✓)	(✓)
Utility 5 (Transparency of Values)	✓	(✓)	✓
Utility 6 (Report Comprehensiveness and Clarity)	(✓)	(✓)	(✓)
Utility 7 (Evaluation Timeliness)	✓	✓	✓
Utility 8 (Evaluation Utilization and Use)	(✓)	(✓)	–
Feasibility 1 (Appropriate Procedures)	(✓)	✓	✓
Feasibility 2 (Diplomatic Conduct)	✓	(✓)	–
Feasibility 3 (Evaluation Efficiency)	(✓)	✓	(✓)
Propriety 1 (Formal Agreement)	–	(✓)	(✓)
Propriety 2 (Protection of Individual Rights)	✓	(✓)	(✓)
Propriety 3 (Complete and Fair Investigation)	(✓)	–	–
Propriety 4 (Unbiased Conduct and Reporting)	(✓)	–	✓
Propriety 5 (Disclosure of Findings)	(✓)	(✓)	–
Accuracy 1 (Description of the Evaluand)	✓	✓	✓
Accuracy 2 (Context Analysis)	✓	(✓)	(✓)
Accuracy 3 (Described Purposes and Procedures)	(✓)	✓	✓
Accuracy 4 (Disclosure of Information Sources)	(✓)	–	–
Accuracy 5 (Valid and Reliable Information)	–	–	(✓)
Accuracy 6 (Systematic Data Review)	–	–	–
Accuracy 7 (Analysis of Qual. and Quant. Information)	–	–	(✓)
Accuracy 8 (Justified Conclusions)	(✓)	✓	(✓)
Accuracy 9 (Meta-Evaluation)	–	(✓)	–

Table 4.2.: Analysis of the SWS Challenge (SWS), S3 Contest (S3) and the WS Challenge (WSC) by Evaluation Requirements

4.4.1. SWS Challenge

Principle difficulties to reliably measure the flexibility of solutions and their adaptability to change, i.e., the software engineering benefits of evaluated semantic technologies are responsible for not meeting the Accuracy 5 and 7 standards. These difficulties are discussed in depth in Section 8.6.4.

A lack of sufficient documentation of the evaluation process and methodologies as well as insufficient documentation of some of the solution approaches results in weaknesses regarding a whole number of standards (Usability 6, Usability 8, Propriety 5, Accuracy 3, Accuracy 4, Accuracy 8 and Accuracy 9), thus stressing the importance of proper, extensive and comprehensible documentation of all aspects of an evaluation.

The scenarios used in the mediation track of the SWS Challenge are, unlike those of the discovery track, not approved by a community consensus process and also not subdivided into smaller subproblems. Both issues are responsible for the identified weaknesses regarding the Utility 4, Feasibility 1, Propriety 3 and Propriety 4 standards.

Finally, responsibilities in the mediation track have not always been clear, especially recently (Propriety 1) and the testbed supporting the evaluation has not always worked in the expected way, was not supported to the greatest possible extent yet and is also not open source such that its debugging and verification is very difficult (Utility 8, Feasibility 3 and Accuracy 6 standards).

4.4.2. S3 Contest

As mentioned previously, the primary Achilles heel of the S3 Contest is its dependency from high quality test data. This is not available in the desired quantity and quality. For some formalisms, no test data is available at all. These problems result in weaknesses regarding the Utility 4, Propriety 3, Propriety 4 and some Accuracy standards.

One of the main points of possible improvement is a more comprehensive reporting of evaluation results. Currently, evaluation results are reported very condensed and briefly. None of the raw data underlying the reported scores is made available. Furthermore, unlike the SWS Challenge and the WS Challenge, the S3 Contest did so far not solicit the collection and publication of technical papers accompanying and describing each entry of the evaluation. All these issues make a detailed analysis and utilization of the evaluation results difficult (Utility 5, Utility 6, Utility 8, Propriety 5, Accuracy 2, Accuracy 4 and Accuracy 9).

The Contest provides a highly useful evaluation environment which greatly reduces the effort involved in participation. Unfortunately, the source code of this environment is not available, which limits its extensibility and make a verification

of the correctness of the data collection and analysis difficult (Utility 8 and Accuracy 6).

Furthermore, the evaluation setup and performance measures used should be discussed more intensively, in particular with respect to their potential pitfalls and limitations and in comparison to alternative measures. These issues are related to significant shortcomings regarding the Accuracy 5 and Accuracy 7 standards.

Finally, more explicit agreements about the responsibilities of people involved in the evaluation and the terms of publication of evaluation results, including options to vet and veto evaluation results before they are released are expected to have a positive impact on the evaluation acceptance (Feasibility 2, Propriety 1 and Propriety 2).

4.4.3. WS Challenge

The WS Challenge reports results only extremely briefly and only for the top performing technologies, not all participants. None of the raw data underlying the scores is made public and the criteria underlying the architectural award are also not provided. This limits the usability and impact of the evaluation results quite significantly. These weaknesses are related to the Utility 6, Utility 8, Propriety 5, Accuracy 2, Accuracy 4, Accuracy 6 and Accuracy 9 standards.

Besides, while the Challenge provides a powerful and useful test data generator, the evaluation fails discussing the limitations of the generated test data or the effects of the test data characteristics on the evaluation results (Utility 4, Accuracy 5, Accuracy 7 and Accuracy 8 standards).

Also the Challenge is really set up as a competitive contest rather than primarily promoting collaboration and cooperative tool improvement. More explicit agreements about the responsibilities of everyone involved and specific terms of publication of evaluation results, including options to vet and veto them before they are published might also help fostering a cooperative atmosphere (Utility 2, Feasibility 2, Propriety 1 and Propriety 2 standards).

Finally, a more extensive discussion of the limitations of the measures used (Accuracy 5 and Accuracy 7), publication of the source code of the evaluation framework (Accuracy 6) and offering an option to participate without the need to attend the evaluation event (Feasibility 3) might further help improving the evaluation quality.

4.5. Conclusions and Delineation of Thesis Scope

The extended analysis presented in the previous sections illustrates that in many ways, research on SWS evaluations is in its beginnings. While there is existing work to build upon, numerous shortcomings and areas of possible improvement exist. In

2008 the candidate together with three experts from the field published an overview journal article on SWS evaluation which concluded as follows:

> "Here is a very brief summary of the status in SWS evaluation as discussed in the previous section:
>
> - With respect to *performance and scalability* on the one hand more and better implementations of matchmakers are needed, on the other hand, standard SWS test collections need to be build.
> - To meaningfully evaluate SWS frameworks' *usability and amount of effort* more fundamental work is needed, in particular, suitable measures on a high level of abstraction need to be identified.
> - Concerning *correctness* what is lacking is a unified approach to evaluating correctness of matchmakers and formalisms, fine grained criteria that are suitable to measure correctness more precisely, and sufficiently large standard test collections.
> - *Coupling* has not been regarded in depth yet, so reliable measures need to be defined. A foundation of those would be, again, standard test collections.
> - *Functional Scope and Level of Automation* are probably the most thoroughly investigated of all the criteria. Nevertheless, to reach meaningful results, a more diverse set of scenarios and a closer analysis of the dependence between scenario, approach, and performance are needed." [KKRPK08]

So far, this thesis contributed a conceptual model for SWS evaluation that provides a framework to relate evaluations to each other, to make their evaluation goals explicit and to meta-evaluate them via a requirements catalogue. The rest of the thesis will be devoted to providing concrete contributions to benchmarking SWS technology. However, the aim of this thesis is not and can not be to provide benchmarking solutions for all evaluation criteria and all possible use cases. An IBM Redbook provides a practitioner's guide to benchmarking and remarks with this respect: "By now you should have realized that with benchmarking we cannot test every possible combination. We must compromise in many areas and use many assumptions. This is why benchmarking is an art and not a science" [HBG+98].

Based upon the analysis provided so far, we now discuss and motivate the choice of the concrete benchmarking contributions provided by the rest of this thesis. The first choice regards the primary SWS task that the contributed benchmarks are primarily concerned with. The focus of this thesis was defined to be on service discovery. This choice reflects personal preference, but also the fact that less existing work than in other areas was available. The WS Challenge, the SWS Challenge mediation track

(that was more developed than the discovery track when this thesis work started) and evaluations from the AI planning community, for instance, are dedicated to providing benchmarks for service composition. Much less work is concerned with service discovery.

It should be noted, however, that the various tasks in service computing (discovery, composition, mediation, negotiation, binding, invocation, monitoring) are closely interrelated. Tasks besides discovery, in particular mediation, composition, binding and invocation will thus be touched and covered in this thesis to some extent, too. Furthermore, some of the methodological work on discovery evaluation is also applicable to the evaluation of other tasks with few adaptations.

Apart from the general SWS task, concrete choices of the evaluation criteria to address and benchmarking problems to tackle had to be made. One major, still lacking prerequisite for meaningful evaluation of SWS frameworks with respect to virtually all criteria are standardized SWS test collections. Addressing this shortcoming was deemed to be fundamental within a general work on SWS evaluation. The following Chapter 5 presents the corresponding contributions. It discusses desirable properties of test collections, presents a framework supporting the collaborative development of standard test collections and presents an initial test collection which complements existing collections in important aspects.

Apart from this, two concrete benchmarks were developed. Chapter 6 will present the work on evaluating the functional scope of SWS frameworks which was primarily performed within the SWS Challenge. As mentioned, the chapter will focus on SWS discovery frameworks, but the approach is also applicable to other application areas like data mediation or service composition planning.

Finally, Chapter 7 will present a setup for evaluating the correctness of SWS discovery frameworks. This setup addresses the before mentioned problems of unifying the evaluation of matchmaker and formalization correctness and includes extensive work on more fine grained, precise and reliable discovery correctness measures. Furthermore it implements a more realistic, decoupled evaluation setting that allows investigating the effects of such decoupling. The evaluation setup was implemented within the S3 Contest.

With respect to evaluation initiatives, this thesis work was performed within the SWS Challenge and the S3 Contest and thus covers both evaluation initiatives in the area that focus on semantic web services and aim at collaboratively advancing the evaluated technologies. The WS Challenge is organized as a competitive contest rather than focusing on collaborative technology improvement and, more importantly, uses a rather low degree of semantics. It is thus less suitable for this thesis work than the other two initiatives.

With respect to criteria, this thesis focuses on *Correctness*, *Coupling* and *Scope and Automation*. As will become evident in the corresponding chapters, *Performance and Scalability* as well as *Usability and Effort* are also covered, but to a

much lesser extent. Evaluations with respect to these two criteria are primarily hampered by practical issues rather than a lack of research on suitable measures or evaluation methodologies. Measures and methodologies for performing performance and usability evaluations have been studied for decades for instance in the areas of databases (performance) or software engineering (usability) and are thus rather well understood. Furthermore, both criteria critically depend on the quality of implementations and the available tool support for a particular technology. Since implementations in the area are currently often prototypical and the available tool support for many approaches is very limited, evaluations with respect to runtime performance and usability seemed less feasible than those with respect to the other criteria.

CHAPTER 5

Test Data for SWS Evaluation

> Imagine what you desire. Will what you imagine. Create what you will.
>
> *(George Bernard Shaw)*

As was discussed in the previous chapter, all benchmarking activities require suitable test data. This chapter discusses requirements on SWS test collections and analyzes the available data. It will be shown that existing data is far from meeting the desirable quality standards. A Web portal will be presented which we have developed to support the collaborative development of better test collections. The chapter concludes with the description of the Jena Geography Dataset, an exemplary test collection that has been developed within this thesis leveraging the above mentioned portal. The work presented in this chapter has been partially published in [KLKR07, KKR08b, KKR08d, KKRK08a, KKRK08b].

5.1. Requirements for SWS Test Collections

As discussed in Section 4.3, serious evaluations of SWS technologies crucially depend on high quality test collections. While the data first and foremost needs to be realistic to resemble relevant real world settings, the precise data needs of an evaluation naturally depend on the evaluation goal being pursued. By recalling the various dimensions of evaluation identified in Section 4.1, a number of desirable characteristics of employed test collections may be derived.

- Performance and Scalability: To compare the performance of different frameworks, they need to run against the same set of services. Thus, offer and goal descriptions in different languages for the same set of services and requests

are needed to compare systems across formalisms. To effectively evaluate the scalability of approaches, a large testbed is needed. However, caution has to be paid with regard to automatically generated testbeds. Depending on how well these reflect the variety encountered in real-world settings, scalability measures may or may not reflect real-world circumstances accurately. The composition of a testbed, for example whether the contained services are very similar to each other or vary greatly, may have strong effects on the performance of certain approaches.

- Usability and Effort: Realistic use case descriptions are required in order to evaluate the difficulty to semantically formalize the corresponding scenarios and encode the involved services and goals within a particular framework. A set of pre-defined ontologies may be provided for some scenarios, while for others the necessary ontologies may need to be created from scratch.

- Correctness and Automation: In order to evaluate the correctness of a framework, real world scenarios need to be addressed with the competing frameworks under evaluation in order to compare the achieved results. The achievable correctness is highly influenced by the expressiveness of the employed formalism and the extent to which the domain of interest is formalized. In particular the latter also always implies a tradeoff between the invested effort and the achieved correctness. Therefore, natural language scenario descriptions are preferable over formal scenario specifications since natural language specifications provide a level playing ground for all approaches and do not dictate the modeling style or degree of formalization being applied. Nevertheless, the provided descriptions need to be as unambiguous as possible, in order to leave as little room as possible for interpretation. Otherwise the results of the evaluation will be tampered by different such interpretations by different people.

- Coupling: The test data and collections need to reflect different people's viewpoints and modeling approaches in order to effectively support testing for this aspect. Test collections developed within one group or even by a single developer do not simulate realistic environments where service descriptions are typically provided in a decoupled way by many independent developers. Thus, preferably many different people should independently contribute to the development of test collections. This will also prevent any unintended bias.

- Functional Scope and Context Assumptions: Investigating and evaluating which approaches support which settings requires a large, but in particular diverse collection of scenarios and services covering as many as possible of the different envisioned use cases. Since different approaches might be advanta-

geous in modeling and processing different types of services, the scenarios need to contain services from different domains and with different characteristics. Generally it is important to keep in mind that the results of any evaluation will depend on assumptions made about the usage context of a use case at hand. Thus, evaluations should be done in different such contexts to make results reliable and universal by preventing them from depending on choices of context parameters. Consequently, test collections need to support different types of services (e.g., information services or transactional services), services from different business settings (e.g., B2C, B2B, or P2P), services with different choreographies (e.g., one step versus complex), etc.

Furthermore, test data should be sufficiently documented to make it reusable in other contexts. This includes technical documentation, e.g., about data formats or necessary libraries, but also information about the characteristics of the data, its sources, the context in which it has been assembled and the procedures that have been applied to create it. Only such information allows to objectively assess whether given data represents realistic data which can be used reliably in contexts different from those for which it has been originally created.

Finally, for many settings, test data should also define specific and concrete requirements about what constitutes a correct solution to the problems represented by the test data. E.g., a test collection to evaluate service discovery not only needs to specify available services and sample service requests, but additionally needs to specify which services should be discovered for a given request. Such specifications must not only be precise and measurable, they again need to state the assumptions under which they have been put together and the procedures that have been applied during their creation. Only this way, their relevance to a given evaluation setting can be independently verified and undergo critical appraisal.

In summary, the ideal test collection needs to be fairly big, be composed of contributions by many different people, cover different domains and realistic services with a variety of characteristics, contain unambiguous natural language descriptions of services as well as semantic descriptions for these services in different formalisms, provide comprehensive documentation and specifications about the tests it has been designed for.

5.2. Publicly Available SWS Test Data

After having discussed the requirements to SWS test data, we will now review the existing publicly available data. We first cover SWS generally visible on the Web and then analyze existing SWS collections explicitly created for evaluation purposes.

5.2.1. Semantic Web Services Visible on the Web

The spread of publicly visible SWS accessible on the Web has been subject of an experimental study by Klusch and Zhing in 2008. They investigated the question "Where are all the semantic Web services today?". They used a specialized crawler to search for Semantic Web Services and found less than seventy links to semantic service descriptions (38 OWL-S, 12 WSML, 11 WSDL-S and 6 SAWSDL) in the surface Web as well as in the scientific archive citeseer [KZ08]. Even if one considers that their search might have only found a minor share of the SWS descriptions available in total, this number is still relatively tiny in comparison to the number of publicly available Semantic Web resources or WSDL files. As of July 2009, the Semantic Web search engine Swoogle indexes almost 3 million Semantic Web documents containing over 700 million RDF triples[1] and the Web Service repository seekda.com provides access to 28434 WSDL files by 7609 providers[2].

Generally, as the authors remark, "the reported preliminary experimental result does not reflect the strong research efforts carried out in the SWS domain world wide in the past few years, independent from the status of maturity of SWS technology and implied low adoption by end users yet." It seems very likely that more SWS descriptions exist than are visible on the surface Web. Klusch and Zhing remark with this respect: "Additional personal communication with few selected research groups at universities [...] revealed that, if semantic Web service descriptions do exist at their site, the public retrieval from specific project related repositories is prohibited, hence invisible to any search engine" [KZ08].

Haniewicz et al. have also recently assessed the current trends and state of the practical SWS adoption by carrying out a market observation, literature studies, analysis of the outcomes and use cases of the various SWS related projects as well as interviews with industry representatives. They come to conclusions similar to those of Klusch and Zhing. "The practical use of Semantic Web services is very low, especially in comparison to the adoption of Web services and their syntactic based interactions. [...] As there are no online repositories, it is hard to find publicly available SWS. Only a few OWLS services descriptions are accessible on the Web" [HKZ08].

Both reports correspond to our own experience. It appears that SWS are caught in a deadlock: SWS will not be widely adopted unless people are convinced that the technologies are mature and improve upon the established state of the art. Proving the technologies mature and validating their benefits requires reliable evaluations. These are infeasible without proper test data, ideally, real data. Real data in turn does not become available until SWS are adopted outside of research. Evaluations and the necessary test data are the leverage at which this thesis tries to break

[1] `http://swoogle.umbc.edu/index.php?option=com_swoogle_stats`
[2] `http://seekda.com/about/about`

Test Collection	Formalism	Size	Released	Comment
OWLS-TC	OWL-S	> 1000	2005	frequently used and updated
SWS-TC	OWL-S	241	2006	not used recently
SAWSDL-TC	SAWSDL/OWL	894	2008	based upon OWLS-TC
Koblenz	DL ($\mathcal{ALC}/\mathcal{ALE}(\mathcal{T})$)	96	2007	based upon OWLS-TC
Assam	OWL-S	164	2004	not used recently
DIANE	NL + DSD	195	2005	not used recently
WS Challenge	WSDL / OWL	n.a.	2008	data generator
SWS Challenge	NL	about 30	2006	complex scenarios

Table 5.1.: Overview of publicly available SWS test collections

this deadlock. With this respect, the surveys quoted above allow drawing two conclusions:

1. There is a need for online SWS repositories that allow sharing and reusing existing SWS descriptions. Such repositories have been lacking. This motivated the creation of a corresponding portal which will be described in Section 5.3.

2. The number of publicly available deployed SWS descriptions is far too small to support meaningful evaluations. Until this changes, SWS evaluations will have to rely on explicitly created test data. The state of the art with respect to such explicit test data will be examined in the following.

5.2.2. Services in Explicitly Created Test Collections

Apart from the few SWS descriptions scattered over the Web, there is also a number of comparably larger publicly available SWS collections that have been explicitly created for evaluation purposes. These are OWLS-TC 3[3], SWS-TC 1.1[4], SAWSDL-TC 1[5], the Semantic Web Service Discovery Data Set by University Koblenz, Germany[6], the Assam collection by Andreas Heß[7], the DIANE Benchmark[8], the WS Challenge data generator[9] and the SWS Challenge scenarios[10]. Table 5.1 shows an overview of these collections among which OWLS-TC is by far the largest and most prominent one. It will be comprehensively discussed, while the other collections will be covered with less detail.

[3] http://projects.semwebcentral.org/projects/owls-tc/
[4] http://projects.semwebcentral.org/projects/sws-tc/
[5] http://projects.semwebcentral.org/projects/sawsdl-tc/
[6] https://www.uni-koblenz.de/FB4/Institutes/IFI/AGStaab/Projects/xmedia/dl-tree.htm
[7] http://www.andreas-hess.info/projects/annotator/owl-ds.html
[8] http://fusion.cs.uni-jena.de/DIANE/benchmark/
[9] http://cec2008.cs.georgetown.edu/wsc08/technical_details.html
[10] http://sws-challenge.org/wiki/index.php/Scenarios

OWLS-TC

Among the collections of semantically annotated services, OWLS-TC is by far the one most frequently used and also regularly cited in the literature. By July 2009, OWLS-TC has had almost 10000 downloads from the Semantic Web tools repository semwebcentral.org[11]. It has been actively developed for several years now and has become the base of two of the other collections (SAWSDL-TC and the Koblenz dataset). It can clearly be viewed as the current de facto standard of SWS test collections. Its popularity is due to its size and the fact that, unlike all other collections (except for the derived SAWSDL-TC), it does not only contain service offer descriptions, but also sample requests and a complete set of binary relevance judgments of the advertisements with respect to the requests. It can thus be readily used in particular to test and evaluate the retrieval performance of OWL-S matchmakers. This was also the purpose for which is has been developed [KFKS05]. Unfortunatley however, OWLS-TC does not provide much information about the conditions under which the relevance judgments have been created. Information about the judges who provided the judgments, the instructions that were given to them and the precise definition of relevance underlying the judgments are largely lacking.

The current OWLS-TC 3 release contains 1007 services written in OWL-S 1.1, roughly half of which are additionally also provided in OWL-S 1.0. Furthermore, it contains 29 OWL-S 1.1 service request descriptions, 28 of which are also provided in OWL-S 1.0. OWLS-TC is organized as a collection of files, each containing a single service description. The OWL-S 1.1 services are classified into seven domains (communication, economy, education, food, medical, travel, and weapon).

Up to OWLS-TC 2.2, binary reference relevance judgments were provided through query folders that for each query contained the set of advertisements judged binary relevant to the query by human judges. Starting with OWLS-TC 3, the relevance judgments are provided by means of an XML file. More importantly, OWLS-TC 3 introduced the usage of graded relevance judgments. This aspect will be discussed in detail in Section 7.5.

Although the manual of OWLS-TC states that part of the services contained in OWLS-TC were retrieved from public IBM UDDI registries, the corresponding original WSDL files are not preserved. However, OWLS-TC 3 contains groundings and corresponding WSDL files which have been generated automatically from the existing OWL-S services via the OWLS2WSDL tool[12].

Despite of the popularity of OWLS-TC, it suffers from a couple of problems and can not be considered a standard test collection [KLKR07]. This is in concordance to the OWLS-TC manual which states: “Please note that no standard test collection for OWL-S service retrieval does exist yet. As a consequence, OWLS-TC can only

[11] `http://www.semwebcentral.org/frs/?group_id=89`
[12] `http://semwebcentral.org/projects/owls2wsdl/`

be considered as one possible starting point for any activity towards achieving such a standard collection by the community as a whole." [KKK+09]. The following critical analysis of OWLS-TC has to be seen in the light of this statement. The criticism covers three aspects.

Use of realistic real-world examples: One common criticism to many use cases and evaluations in the service matchmaking domain is the use of artificial toy examples which are far from realistic applications. Even though examples do not necessarily have to be realistic to test features of a matchmaking system, the use of real-world examples clearly increases the relevance of an evaluation and reduces the danger of badly designed test data. Furthermore, toy examples far from real-world applications generally hinder the acceptance of new technology by industry. Unfortunately, a substantial share of the OWLS-TC services seems artificial and somewhat idiosyncratic. Some services suffer from obvious copy and paste errors. Sometimes the semantics of the services is incomprehensible even for a human expert. This is aggravated by the fact that, as mentioned, no information about the original sources of the services has been preserved.

A comprehensive coverage is impossible due to the size of OWLS-TC but the following examples illustrate the issues (in the following, service names always refer to the name of the corresponding service description file, not the service name from the service's profile, quotes are from the service's description files):

- Some services are simply erroneous, a few services, for instance, are pair wise identical except for the informal textual description (e.g. `_price_CannonCameraservice.owls` and `_price_Fishservice.owls`)
- The service `__destination_MyOfficeservice.owls` is supposed to "return destination of my office", but takes concepts of type *organization* and *surfing* (which is a subclass of *sports*) as input.
- The service `surfing_farmland_service.owls` is described as "This is the recommended service to know about the farmland for surfing" and has an input of type *surfing* and an output of type *farmland*. What's the semantic of this service?
- The service `qualitymaxprice_cola_service.owls` "provides a cola for the maximum price and quality. The quality is an optional input." It is described by its inputs of type *maxprice* and *quality* and an output of type *cola*. There are numerous similar services that return cola (six more services), beer + cola, coffee + whiskey (eleven services), cola-beer, cola + bread or biscuit (two services), drinks (three services), liquid, whiskey + cola-beer as well as irish coffee + cola. The semantics of these services remain unclear.

- The service `UnsuccessfulDiagnosis_service.owls` "informs you about the diagnostic process, that is proved unsuccessful, with reasoning." It has outputs of type *DiagnosticProcess* and *Reasoning* and no input. What does this mean?

Semantic richness of descriptions: Services should not only be realistic and realistically complex, their semantic descriptions should also be sufficiently precise to potentially allow showcasing the powers of sophisticated semantic technologies. After all there should be an advantage using semantic annotations compared to simply using traditional information retrieval techniques. Unfortunately, the services of OWLS-TC are described rather superficially. First of all, all services are solely described by their inputs and outputs. The descriptions do not make use of more advanced concepts like preconditions or effects. However, what is the semantic of a service (`car_price_service.owls`) that takes a concept of type *Car* as input and has a concept of type *Price* as output? It might sell you a car and tell you the price afterwards, it might just as well only inform you about the price of a new car or the price of a used car. It might rent a car for the returned price. It might tell you the price of the yearly inspection for the given car. There are many different possible interpretations. What is the semantic of a service like `car_priceauto_service.owls` that takes as input a concept of type *Car* and has outputs of type *Price* and *Auto* (which is a subclass of car)? Such ambiguities remain unresolved in OWLS-TC.

Generally, the textual descriptions of the service offers and queries are sometimes not captured well by the semantic descriptions. Query 23, for instance, is informally described as "the client wants to travel from Frankfurt to Berlin, that's why it puts a request to find a map to locate a route from Frankfurt to Berlin." This request is described (`geographicalregiongeographical-region_map_service.owls`) as a request for a service with two unordered inputs of type *geographical region* and a single output of type *map*. Clearly routing services will also be found (among many others) by such a request, but for many use cases such descriptions may not necessarily allow to demonstrate the added value of *semantic* service discovery.

Independence of offer and request descriptions: Ideally, service offer and request descriptions should be designed independently since this is the envisioned situation in reality. Service providers describe their offers, clients query for a service with a semantic request description and a matchmaker is supposed to find the offers that match the request. In laboratory settings it is sometimes desirable to artificially design the offers to match a request at various degrees. This ensures that all potentially existing degrees of match occur during a test run. However, a test where the offers have been designed to match a request at hand with specific degrees runs the risk of doing nothing more than supporting the belief that a particular matchmaker *implementation* operates as expected. It does not demonstrate the power of some

semantic description formalism or a certain matchmaking approach. Unfortunately, it appears that a substantial share of the OWLS-TC services have been created through variations of existing services with a given query in mind. Query 4 for instance asks for the combined price of a car and a bicycle. It seems quite idiosyncratic to buy a car and a bicycle as a package, yet there are at least eleven service offers in OWLS-TC that precisely offer to provide the price of a package of one car and one bicycle.

Conclusions: OWLS-TC has to be acknowledged as the largest, the by far most popular and also the first publicly available SWS test collection. Many of the mentioned criticisms still origin from early versions of OWLS-TC and remain in current versions due to the tremendous effort involved in developing and maintaining large numbers of semantic service descriptions just for testing purposes. The critics given above should be understood in light of these considerations. Nevertheless the given examples highlight problems in OWLS-TC that will have to be overcome before OWLS-TC can be truly considered the standard test collection as which it currently serves.

SWS-TC

SWS-TC 1.1 is a smaller dataset of 241 OWL-S service descriptions developed by Yasser Ganjisaffar and Hadi Saboohi for the evaluation of a similarity measure for OWL-S Web services in 2006 [GANJ06]. Since it original publication, SWS-TC has not been updated and it does not appear to be used anymore.

Compared to OWLS-TC, the level of documentation available for each service seems to be a bit higher. Furthermore, the services of SWS-TC seem to be somewhat more realistic than those of OWLS-TC. Unlike OWLS-TC, which relies on several different ontologies, the SWS-TC 1.1 descriptions are based on a single unified domain ontology. Apart from that, the descriptions are similar in nature to those from OWLS-TC. Most importantly, the services of SWS-TC are also exclusively described via their interface and do not formalize preconditions or effects. As is the case with OWLS-TC, the original WSDLs of the services that underlie SWS-TC have not been preserved and are not available anymore. Furthermore, SWS-TC does not provide sample queries or reference relevance judgments.

SAWSDL-TC

SAWSDL-TC is a semi-automatic translation of a subset of OWLS-TC 2.2 from OWL-S to SAWSDL via the OWLS2WSDL tool[13]: “OWLS2WSDL transforms

[13] `http://projects.semwebcentral.org/projects/owls2wsdl/`

OWLS service descriptions (and concept definitions relevant for parameter description) to WSDL through syntactic transformation. Top-level annotations taken from the original OWL-S descriptions have been added for XML Schema type definitions used to describe message inputs and output" [KKZ09].

The current first release of SAWSDL-TC (published July 2008) contains 894 semantically annotated WSDL service offer descriptions and 26 semantically annotated WSDL request descriptions. Additionally, 1607 XSLT files with automatically generated lifting schema mappings are provided.

With respect to the options that the SAWSDL standard offers, SAWSDL-TC is currently limited as follows [KFKK08]:

- Only one single interface per description file.
- One one single operation per interface.
- Model references point to concepts described in OWL-DL exclusively.
- Only automatically derived lifting schema mappings in XSLT are provided.

Being a semi-automated translation, SAWSDL-TC inherits all main characteristics (service characteristics, ontologies used, relevance judgments provided, ...) from OWLS-TC. Similar to OWLS-TC, SAWSDL-TC is frequently used and cited in the literature.

Koblenz Dataset

The Semantic Web Service Discovery Data Set by University Koblenz, Germany, is a set of Semantic Web Services described by the use of a DL-based framework proposed by Grimm et al. [GMP04]. It has been derived from services of OWLS-TC and been built and used as test collection for performing a clustering-based service discovery process [dSFE07]. "It consists of an $\mathcal{ALC}$ ontology representing the knowledge base of reference and a set of $\mathcal{ALE}(\mathcal{T})$ services described using such an ontology. The ontology models broad domains: bank domain, post domain, means of communication domain and geographical information. On the ground of such an ontology, 96 complex concept descriptions acting as service descriptions have been built" [dSFE07]. To the best of our knowledge, the Koblenz Dataset has not been used outside of the context for which it was developed.

ASSAM Collection

Andreas Heß and Nick Kushmerick have built two test collections for their ASSAM WSDL Annotator[14]: a collection of categorized WSDL services[15] gathered

[14] `http://www.andreas-hess.info/projects/annotator/`
[15] `http://www.andreas-hess.info/projects/annotator/ws2003.html`

from xmethods.com and salcentral.com and a collection of 164 OWL-S services[16] which has been built by annotating a subset of the services from the WSDL collection [HJK04]. The collections are quite old (from 2004) but still available. The service modeling style corresponds to that used in OWLS-TC and SWS-TC. The ASSAM collection is the only one that provides the original WSDL files together with the semantic descriptions. While the WSDL collection has been reused for the evaluation of service discovery approaches based on information retrieval techniques like [SW05, KvdHD06a], it appears that the OWL-S collection has never been reused.

DIANE Benchmark

The DIANE Benchmark defines tests to experimentally evaluate the expressiveness of SWS description formalisms. The approach is to define natural language requests, which have to be formalized in the SWS formalism of interests. SWS experts then assess which share of the requests could be properly expressed with the given formalism. Unfortunately, the benchmark does not specify concrete objective requirements to base this assessment on.

The benchmark distinguishes between end-user requests asking for a specific functionality that needs to be provided ad-hoc and application requests asking for Web services in order to embed them in a service-oriented application. To define both types of queries, a group of test subjects not familiar with Semantic Web technology was asked to formulate natural language service requests for three different application domains. For the end-user requests, 100 book buying requests and 45 train ticket requests were formulated. For the application requests, 50 queries from the tourism / trip planning domain were created. The 195 queries are available in English natural language text. Furthermore, the benchmark has been applied to the DIANE framework [KKRKS08]. From this application, semantic descriptions for the queries are available in the non-standard formalism DSD [KKRM05]. We are not aware of any application of the benchmark other than the DIANE evaluation.

WS Challenge Test Data Generator

The WS Challenge has published test data generators for its 2008 and 2009 editions[17] [BWG09]. These generators are able to create large test corpora of artificial WSDL descriptions. The WSDLs contain references to semantic definitions for the input and output messages in an OWL ontology. However, the used OWL ontology is effectively restricted to the expression of inheritance relationships, thus, the

[16] `http://www.andreas-hess.info/projects/annotator/owl-ds.html`

[17] Available at `http://cec2008.cs.georgetown.edu/wsc08/technical_details.html` and `http://ws-challenge.georgetown.edu/wsc09/technical_details.html`

level of semantics employed is much lower than typically in SWS settings. Notably, the generated WSDLs also do not employ the SAWSDL standard but use a custom WSDL extension. This hinders the reuse of the generators outside of the WS Challenge. The WS Challenge presents a composition challenge and includes precise specifications about what constitutes a correct composition as well as quality measures to further assess the relative quality of correct compositions.

SWS Challenge Scenarios

As discussed in Section 3.1.1, the SWS Challenge defines a set of complex scenarios to evaluate SWS frameworks. These scenarios differ from the above described datasets in that they describe rather complex and detailed use case scenarios instead of collecting large numbers of services or service descriptions. This corresponds to the different scope of the Challenge, which certifies the functional scope of technologies rather than providing quantitative measurements of, for instance, retrieval precision or runtime performance. The scenarios contain specifications about what constitues a correct solution. The evaluation approach implemented in the SWS Challenge is covered in detail in Chapter 6.

To enhance the SWS Challenge testbed, a scenario proposal and reviewing process for scenario contribution is in place. Existing scenarios have been contributed by different groups. They are described in English natural language and mostly backed by executable service implementations for which WSDL descriptions are available. Altogether, the scenarios involve around 20 services and 30 service requests. Even though the scenarios have been implemented and solved by a number of groups, not all solutions, in particular only few semantic service descriptions are available. The SWS Challenge scenarios are described in detail in Section 6.4.

5.2.3. Conclusions

Recalling the summary of the requirements to test collections, the ideal test collection needs to be fairly big, be composed of contributions by many different people, cover different domains and realistic services with a variety of characteristics, contain both, unambiguous, natural language descriptions of services and semantic descriptions for these services in different formalisms and finally provide comprehensive documentation and specifications about the tests it has been designed for. Table 5.2 shows an assessment of the discussed collections with respect to these requirements. A checkmark "✓" denotes a fulfilled requirement, a checkmark in parentheses "(✓)" a partially fulfilled requirement and a dash "–" a requirement that is not fulfilled. The table illustrates that, unfortunately, all collections are far from meeting all requirements yet:

	OWLS-TC	SWS-TC	SAWSDL-TC	Koblenz	ASSAM	DIANE	WS Challenge	SWS Challenge
Large size	✓	(✓)	✓	–	(✓)	(✓)	✓	–
Contributions by many groups	(✓)	–	–	–	–	–	–	✓
Covers different domains	✓	✓	✓	(✓)	✓	(✓)	–	(✓)
Realistic services	(✓)	✓	(✓)	(✓)	✓	(✓)	–	✓
Variety of characteristics	–	–	–	–	–	(✓)	–	(✓)
NL descriptions available	(✓)	(✓)	–	–	–	✓	–	✓
WSDLs available	–	–	✓	–	✓	–	✓	✓
Original real WSDLs available	–	–	–	–	✓	–	–	–
Number of formalisms	2	1	2	1	1	1	1	?
Comprehensive documentation	–	–	–	–	(✓)	(✓)	(✓)	(✓)
Specification of intended tests	(✓)	–	(✓)	–	–	(✓)	✓	✓

Table 5.2.: Assessment of publicly available SWS test collections with respect to test collection requirements

- Only OWLS-TC, SAWSDL-TC and the data generated by the WS Challenge data generator are large datasets with around 1000 or more services. SWS-TC, The ASSAM collection and the DIANE Benchmark specify around 200 services and the remaining collections define less than 100 services.

- Apart from the SWS Challenge scenarios and, to a lesser degree, OWLS-TC, all collections have been developed within a single research group.

- About half of the collections are specific to narrow domains and almost none contains services with truly varying characteristics. Even though OWLS-TC, for instance, contains a large number of services from different domains, the descriptions all share relatively similar characteristics with respect to their complexity, the number of parameters, their choreography, etc. The SWS Challenge and the DIANE Benchmark define somewhat more varied services, but have the drawback of their comparatively much smaller sizes.

- The DIANE Benchmark and the SWS Challenge scenarios are the only collections that contain natural language, i.e., formalism independent, descriptions of the contained services. However, OWLS-TC and SWS-TC contain some natural language comments in their descriptions.

- Even though all collections except for the WS Challenge generator and the DIANE Benchmark (which focuses on queries instead of services) claim that they are modeled after real services, only the ASSAM dataset contains the original WSDL descriptions. For all other collections, it is not verifiable whether the service semantics are appropriately captured by the provided descriptions and whether the descriptions indeed reflect realistic services.

- None of the collections allows evaluation across formalisms. Since SAWSDL-TC is a translation of OWLS-TC, these collections can be assumed to be available in two formalisms (OWL-S and SAWSDL), albeit with the before mentioned limitations with respect to SAWSDL-TC and the SAWSDL standard. The SWS Challenge scenarios have been solved using different formalisms and technologies. Unfortunately, the corresponding solutions and semantic descriptions are not readily available. All other collections are entirely specific to one formalism.

- Existing collections are generally poorly documented. None of the collection contains really comprehensive information. In particular information about the source of the services and the processes and conditions under which they have been selected or formalized are almost completely lacking.

- Only the WS Challenge and the SWS Challenge provide fairly precise specifications about the tests that these datasets have been designed for. OWLS-TC and SAWSDL-TC provide specific information about the tests, but lack comprehensive information about how the provided gold standard (represented by the reference relevance judgments) has been obtained. Similarly, the DIANE Benchmark defines tests, but at least some of these are based on rather subjective judgments and lack objective, measureable definitions. The other datasets do not define any tests to be performed with them at all.

Furthermore, there are a number of additional problems to observe. Despite of significant research efforts outside the OWL-S community, there is little publicly available test data for tools and algorithms relying on formalisms other than OWL-S, in particular, there is no publicly available test data for WSML/WSMO.

The existing collections tend to be rather poorly structured and documented and thus difficult to use, in particular difficult to reuse. Apart from the WS Challenge generator, to which this criteria does not really apply, and the SWS Challenge, which provides its scenario descriptions in a fairly well maintained wiki, all collections come as archives of flat files that are difficult to browse, search and process. This also probably explains why some datasets (SWS-TC or the ASSAM collection) have been rarely reused despite of being available in a wide-spread formalism.

Most importantly, except for the SWS Challenge scenarios, none of the collections provide an easy to use, well-defined mechanism for updates or contributions. The SWS Challenge scenarios may be updated by everyone on the corresponding wiki and there is a mechanism in place for new scenario proposals. For all other collections, the only way to contribute is to personally contact the original author of the collection. Apparently, this has been an obstacle for more community participation in the development of existing collections.

Building SWS test collection involves a tremendous amount of effort, since descriptions have to be built explicitly and much of this work needs to be done manually. Therefore, building large and high quality collections requires more effort that can be supplied by any single group. Even if there were a particularly resource-rich group, the need for impartiality would still make it undesirable to have this one group develop a standard test collection alone. As a consequence, community involvement is crucial for successfully building high-quality test collections. Feitelson remarks with this respect:

> "Getting data is hard. Getting good data is even harder. It is therefore imperative that data be shared, so that the most benefit possible will be gleaned from it. In particular, sharing data enables two important things: 1. Exploration – there is always more to the data than you initially see. By making it available, you enable others to look at it too. Paraphrasing Linus's Law, with enough eyeballs, the data will eventually give up its secrets. 2. Reproducibility – given your data, others can redo your analysis and validate it." [Fei06]

One prerequisite for sharing data and obtaining contributions from the community is to offer appropriate tools that make contributing as easy and effortless as possible while offering a significant and obvious gain. Such tools have been largely lacking so far. In the following section, we will describe our contribution targeted at resolving this problem.

5.3. OPOSSum: Tool Support for Community Involvement

In order to provide the necessary tool support making the collaborative development of standard SWS test collections feasible, we have developed OPOSSum, the Online POrtal for Semantic Services[18]. We will motivate and explain the design of OPOSSum and provide information about its implementation.

[18] `http://fusion.cs.uni-jena.de/OPOSSum`

5.3.1. Design Goals

According to the discussion in the previous sections, the design of OPOSSum has been motivated by three main objectives.

Goal 1: Promote exchange, reuse and collaborative improvement of existing data. As mentioned above, there must be many more SWS descriptions around than were found by the experiment by Klusch and Zhing. Despite of major projects in the field in Canada, Asia, or the pacific rim, for instance, Klusch and Zhing did not find any public semantic web services outside of the US, Europe and Iran. Apparently, most SWS descriptions developed within research projects remain hidden in private repositories. Similarly, three of the test collections introduced in the previous section are available on project web sites only and thus hard to find.

Existing data has to be shared more efficiently for two reasons. First, the amount of effort involved in creating SWS descriptions simply requires reusing existing work. Second, using independent third-party data is also essential to avoid unintended biases and increase the objectivity and thus relevance of an evaluation. Therefore, sharing, reusing, and editing existing data must become easy.

Goal 2: Improve structure, documentation, and usability. As discussed, the few public as well as private SWS collections that we know of are generally poorly documented, poorly structured, and usually come in form of collections of flat files, which do not support convenient browsing or powerful search. This limits their usability and keeps people from actually reusing them. Future collections must be improved in this aspect. To make this happen, this must be supported by tools.

Goal 3: Support reuse and comparisons across formalisms. Besides the effort to semantically annotate a given service, it is also far from trivial to come up with meaningful, rich, and diverse services in the first place (one of the obvious lessons learned from existing collections). Building a SWS test collection requires to gather (potentially fictitious) services and to semantically annotate them. Both steps are similarly challenging and time-consuming. Thus a collection of meaningful services in any description format including natural language is of great value when constructing a collection in a particular formalism. Furthermore, testbeds for different SWS description formalisms should not be isolated, as they currently are, but instead be closely interlinked. This supports the direct comparison of different description approaches for the same set of services, thereby allowing to investigate the trade-offs of the various approaches more easily. Therefore, future collections need to support the management of different descriptions in different formalisms for the same set of services.

5.3.2. Data Model

The internal data model of OPOSSum has been designed to support the three goals listed above. As a result of Goal 2, unlike existing file-centered collections, OPOSSum is built on top of a relational database. Figure 5.1 shows a slightly simplified picture of the data model of OPOSSum's database.

Unlike all existing collections, OPOSSum's data is structured around the notion of a *Service*, independent from a particular service description. This promotes the reuse and comparison of service descriptions written in different formalisms (Goal 3). Accordingly, a service in OPOSSum is first described by a natural language text. Services can be classified in possibly overlapping *Categories*.

To add more structure and support more precise searching, a service's *Parameters* (inputs and outputs) should be declared explicitly and described in natural language in addition to the general description of the service (Goal 2). To add more semantics without binding to a particular formalism, parameter types are mapped to WordNet synsets. WordNet[19] is a semantic lexicon for the English language developed at Princeton University. It uses the notion of synsets to collect synonyms and disambiguate homonyms. Sense keys are used to reference synsets, thus providing an unambiguous identifier of a particular semantic meaning. Using WordNet synsets provides a kind of semantics that ensures an excellent compromise between being unambiguous, flexible, easily usable and language/formalism independent. We report on some experiences using WordNet in Section 5.3.4.

While some approaches to SWS model service offers and requests alike (e.g., OWL-S), others model them differently (e.g., WSMO). This enables to distinguish between more generic offers (like a flight booking service) and concrete requests (like a booking request for a particular flight). To accomodate both views, OPOSSum explicitly distinguishes between service *Offers* and *Requests*.

An arbitrary number of pointers to *Service implementations* (e.g., a web service) may be listed for any OPOSSum service.

Service descriptions written in any *Formalism* (WSDL, OWL-S, SAWSDL, WSML, ...) are collected by attaching them to the service (request or offer) that they describe. As mentioned above, this should ease the creation of descriptions in different formalisms and support the comparisons of different descriptions for the same service (Goal 3).

Services as well as Service descriptions may be grouped to *Service collections*.

Resources like ontologies or schemas can be added to the system as independent entities. Resources and of course descriptions may refer to the resources that they import. This allows to transitively compute the set of necessary resources for a given set of service descriptions and thus to automatically assemble test collections with all necessary resources on the fly.

[19] `http://wordnet.princeton.edu/`

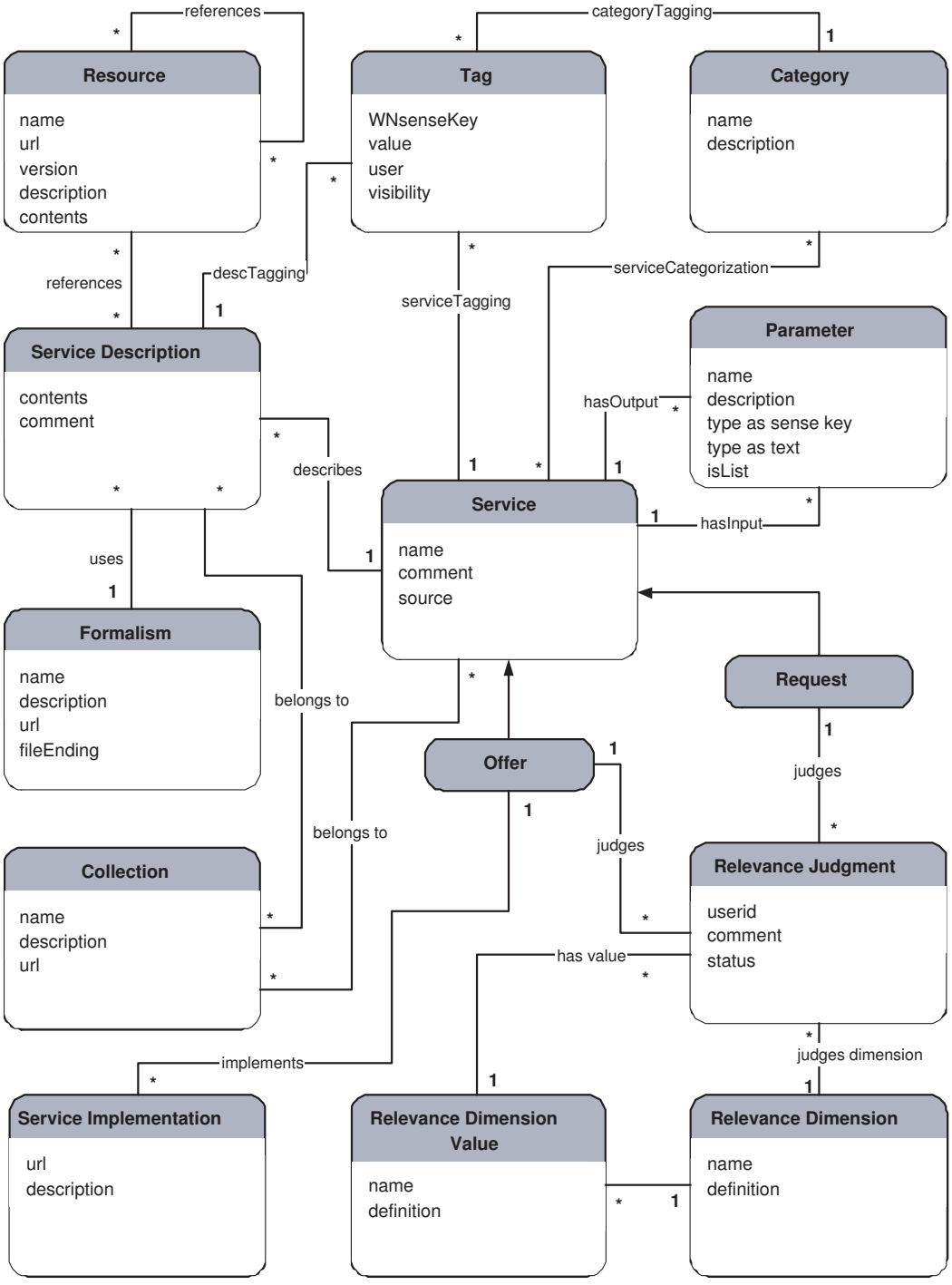

Figure 5.1.: Relational data model of OPOSSum

To enable users to mark services, OPOSSum allows adding *Tags* to services as well as service descriptions in a Web 2.0 fashion. To support more powerful searching, those tags may optionally be linked to WordNet sense keys to disambiguate their semantics. OPOSSum supports private as well as public tags.

Finally, most evaluations of SWS retrieval require to obtain reference *Relevance judgments* that assess the relevance of a service offer to a service request (see Chapter 7). For maximum flexibility, OPOSSum supports multi-dimensional, graded, user-based relevance judgments. This means that:

- Relevance between a service request and a service offer can be assessed according to different relevance definitions (called *Relevance dimensions*) in parallel. The available relevance definitions can be easily configured.
- Each relevance dimension supports a configurable set of arbitrarily many *Relevance dimension values*.
- Each user may provide an own, commented judgment according to each relevance dimension. This allows creating redundant judgments to check for judgment consistency.

Relevance judgments and their management using OPOSSum will be discussed comprehensively in Chapter 7.

5.3.3. Implementation and Status

To enable easy sharing, reusing, and editing of existing data (Goal 1), OPOSSum has been implemented as a PHP-based web portal on top of a MySQL database. It is accessible online at `http://fusion.cs.uni-jena.de/OPOSSum`. This way – unlike with the existing collections – anybody willing can easily contribute to the collection hosted in OPOSSum in a wiki-like fashion. OPOSSum is fully functional. Users may search for services or descriptions based on their properties, compose and download collections based on search results, add new services and service descriptions to the database, store relevance judgments for offers and requests, improve the documentation of existing data, update descriptions, fix errors and inconsistencies in referenced ontologies, etc. Figure 5.2 shows a screenshot of the portal.

OPOSSum has been presented at the European Semantic Web Conference 2008 and the International Conference on Semantic Computing 2008, where it received the Best Demonstration Award [KKRK08a, KKRK08b]. The most important existing SWS test collections have been fully integrated with OPOSSum. Details on this integration will be given in the following section. As of January 2010, OPOSSum has 44 registered users and lists 2851 descriptions for 1524 services. This makes it the by far largest collection of SWS test data available.

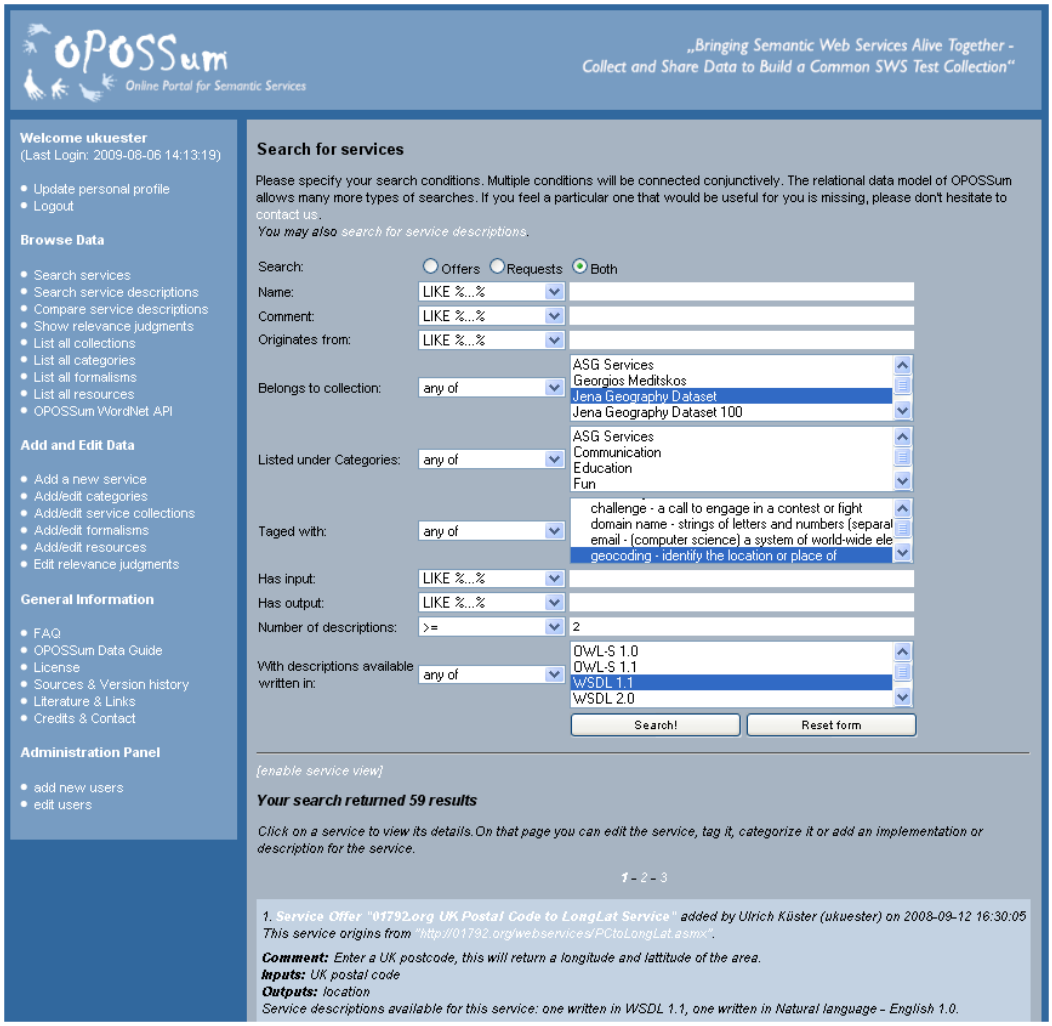

Figure 5.2.: Screenshot of the OPOSSum Portal

5.3.4. Integration of Existing Data with OPOSSum

To leverage existing work we completely imported OWLS-TC, SWS-TC, and SAWSDL-TC to OPOSSum. In this section, we will present some more information about the imported data and describe how it has been integrated with OPOSSum.

OWLS-TC

As mentioned previously, OWLS-TC is by far the largest and best established publicly available collection of semantic web services. The imported 2.2 version of OWLS-TC contains 1003 services written in OWL-S 1.1, 69 of which are also provided in OWL-S 1.0. Additionally, it contains 29 OWL-S 1.1 service request descriptions, 28 of which are also provided in OWL-S 1.0. OWLS-TC is organized as a collection of files, each containing a single service description. The OWL-S 1.1 services are classified into seven domains (communication, economy, education, food, medical, travel, and weapon). Binary reference relevance judgments are provided through query folders that contain the set of advertisements judged relevant to the corresponding query by human judges. We parsed the service offer and query description files and extracted the following data from each of them:

- domain the service was listed under
- resources (ontologies) imported by each description
- service name as specified in the profile
- textual description as specified in the profile
- service inputs and outputs:
  - parameter name
  - parameter type (as an ontological reference)
  - parameter description as specified in the label attribute of a parameter (although next to none were provided).

Overall, the 1032 services used 3020 parameters with 417 different parameter types. These were manually mapped to WordNet sense keys as used in OPOSSum. The resulting type mapping is available online[20]. During the parsing and the process of mapping the parameter types, we noticed a few errors in the collection that we corrected. A listing of the corresponding changes is also available online[21].

[20] `http://fusion.cs.uni-jena.de/OPOSSum/docs/OWLSTCMappings.txt`
[21] `http://fusion.cs.uni-jena.de/OPOSSum/index.php?action=faqOWLSTC`

From the data obtained from each of the descriptions we inserted a corresponding service offer or request in OPOSSum and attached the original description for that service to the new service entry. The ontologies imported by those descriptions were inserted as well and linked to the descriptions that referenced them. The OWL-S 1.0 service descriptions additionally contained in OWLS-TC 2.2 were later added in the same way to the corresponding service entry. All generated service entries were categorized according to the domains under which they were listed in OWLS-TC 2.2. As a last step, the OWLS-TC relevance set folders were parsed and the explicit positive as well as the implicit negative relevance judgments were added to OPOSSum's database.

OWLS-TC was thus completely integrated, but unfortunately the generated entries do contain errors, mostly related either to incorrect mappings of parameter types to WordNet synsets or to flaws in the original OWLS-TC 2.2 data discussed in Section 5.2.2.

Some incorrect mappings of parameter types to WordNet synsets result from the fact that those mappings were made based on the referenced ontological types independent from the actual services. As a result they do not always describe the parameter types very well. First, the ontological types did not always have a perfectly matching entry in WordNet. Second, the ontological types themselves did not always capture the semantics of the parameter types precisely in the first place. Obviously, these two sources of errors can potentially combine in an unfortunate way. We report on some of the problems encountered when creating the WordNet mappings below. The automatically generated service entries were added to a specific category ("Incomplete - OWLS-TC") to mark that they could be considerably improved, if a human took some time to manually edit them. However, due to the size of OWLS-TC 2.2 this can only be done by the community as a whole. It is one of the achievements of OPOSSum that this is now easily possible online.

OWLS-TC 2.2 contains roughly two dozen different domain ontologies that are referenced by the services. During the process of creating the type mappings, we realized that these contain concepts which are defined multiple times in the different ontologies without being explicitly related to each other. For some settings, this situation may be more realistic than one with a single unified ontology, thus, this is not a flaw of OWLS-TC, but it is a feature worth noting.

SWS-TC

SWS-TC 1.1 is the second largest test collection of 241 OWL-S services. While SWS-TC 1.1 is much smaller than OWLS-TC, it seems that the average quality of the descriptions is somewhat better (e.g., the level of documentation) and that it contains fewer services created by only slight variations of existing ones.

The services from SWS-TC 1.1 have been parsed and added to OPOSSum exactly like those from OWLS-TC 2.2. The WordNet mappings created for the SWS-TC 1.1 service's parameter types are also available online[22]. Overall, the 241 services in SWS-TC 1.1 used a total of 594 parameters with 192 different parameter types, which illustrates the relatively higher variety of these services compared to OWLS-TC 2.2. Unlike OWLS-TC 2.2, SWS-TC 1.1 is based on a single unified domain ontology. Services in SWS-TC 1.1 are not classified in domains. Therefore, after their insertion to OPOSSum, a volunteer classified them manually in 12 overlapping categories (in contrast, the categorizations of the services from OWLS-TC 2.2 derived from the domain classification in that collection are disjoint). Neither sample requests nor relevance judgments are available for the SWS-TC services.

SAWSDL-TC

SAWSDL-TC is a semi-automatic translation of OWLS-TC 2.2 from OWL-S to SAWSDL. The current first release contains 894 semantically annotated WSDL service offer descriptions and 26 semantically annotated WSDL request descriptions. Additionally, 1607 XSLT files with automatically generated lifting schema mappings are provided. Since SAWSDL-TC is a translation of OWLS-TC and the described services had thus already been imported to OPOSSum, the descriptions from SAWSDL-TC could easily be integrated with OPOSSum by simply attaching them automatically as alternative descriptions to the corresponding existing service entry. OPOSSum allows arbitrary resources to be references by service descriptions. However, to keep the listed data as simple as possible the 1607 XSLT files were not added as independent resources. Instead, the contained XSLTs were copied into the corresponding WSDL files and the SAWSDL liftingSchemaMapping references were changed to local references within the WSDL file.

Since most additional data like service classifications or reference judgments is handled in OPOSSum on the level of a service entry and not on the level of service descriptions, all of the corresponding information from SAWSDL-TC was already contained in OPOSSum and no further work was necessary to finish the import of SAWSDL-TC.

Mapping Parameter Types to WordNet Sense Keys

The main task to integrate OWLS-TC, SWS-TC, and SAWSDL-TC to OPOSSum consisted of mapping the service's parameter types to WordNet sense keys. These mappings had to be created manually since this information was not contained in the original releases but constitutes an improvement over the original collections. When creating the mappings, we looked up the concepts in the ontologies, tried

[22] `http://fusion.cs.uni-jena.de/OPOSSum/docs/SWSTCMappings.txt`

to estimate an English term which captured their semantics best and identified the corresponding WordNet synset using OPOSSum's WordNet API[23]. Although WordNet is likely the most complete and thorough ontology available, a couple of terms were difficult to map to WordNet synsets:

- Many compound terms are missing in WordNet, even though some of them are extremely common, e.g. email address, credit card number, account name, user name, dvd player, airport code, airline code, area code, arithmetic computation, full-time respectively part-time position, ...
- Many technical terms are missing, too: SLR (single lense reflex camera), APS (advanced photo system), analog SLR, ...
- A common feature in ontologies is a hierarchy of concepts where general concepts are specialized by adding restrictions. Such concepts generally did not map well to WordNet which often contains only the base concept. Examples are: cell phone with camera, recommended price, price in Euro, price in Dollar, tax free price, taxed price, ...

In particular the absence of some very common compound terms (like email address or credit card number) is astonishing. It seems that Wordnet is surprisingly weak in this aspect. In some cases, however, it may also be that we simply did not find the best matching term. The OPOSSum WordNet API performs a quite basic search in the WordNet dictionary, thus, it is often not trivial to come up with the right search term. The term "CryptographyKey" used in SWS-TC's concepts ontology for instance, was mapped to the WordNet sense 106492320 ("a word that is used as a pattern to decode an encrypted message") which is listed under the lemma "key word". At the time the mappings were created, a search for "key" listed 21 senses but did not list that particular one because it did not perform a substring search if precise matches were found. Also the search terms "cryptographic" or "cryptography" do not yield the above mentioned sense. The OPOSSum's Wordnet API has meanwhile been improved to also support substring search.

Other Collections

While OWLS-TC, SWS-TC, and SAWSDL-TC have been completely integrated with OPOSSum, the other collections mentioned in Section 5.2.2 have not been integrated yet. The main difficulties are as follows.

The Koblenz database consists of a single RDF-serialized ontology which would have to be split in descriptions for the single services in order to add these to OPOSSum. Furthermore, it does not contain any natural language documentation

[23] `http://fusion.cs.uni-jena.de/OPOSSum/index.php?action=wordnetAPI`

regarding the semantics of the services. In order to add meaningful service entries to OPOSSum, those semantics would have to be derived manually from the description logic formalization of the services which is a very time-consuming task.

Somewhat the opposite is the case for the DIANE Benchmark. It consists primarily of natural language descriptions of the services, but formal descriptions are only available in a graphical format which can not be parsed easily. This makes it very hard to extract certain structure, like the inputs and outputs of the services. Again, these would have to be added completely manually.

The Assam collection appears to be severely outdated and many of the referenced ontologies are not available at the specified URLs. These ontologies would have to be manually replaced by new ontologies before the collection can be used.

5.4. The Jena Geography Test Collection

OPOSSum integrates the existing test collections in one place and allows easily updating and improving them. While this constitutes an important step towards better test collections and the integration of previously separated datasets is an improvement already, the data is still far from meeting the requirements for standard SWS test collections. In particular, it is still unclear, whether the data really corresponds to real or at least realistic services, the natural language description of the imported data is very scarce and there are no alternative descriptions in a variety of formalisms for the same set of services available. In this section we report about the Jena Geography Dataset (JGD)[24], a new collection we built that comes closer to meeting the requirements identified in Section 5.1.

One of the difficulties when building a test collection of SWS is to ensure for a sufficient variety of services and, at the same time, make sure that the contained services are still somewhat related and similar. For many evaluation purposes it is particularly important to have services which are similar to each other in varying degrees, i.e. services which are very similar to each other, others that are completely different, and as many in between as possible.

We therefore examined public web service repositories to identify a domain with many public services that meet this requirement. For the start we chose the geography domain since it appears to be the domain with the most publicly accessible web services available. We analyzed and collected over 200 service operations within this domain from sources like `http://seekda.com`, `http://xmethods.com`, `http://webservicelist.com/`, `http://www.programmableweb.com/`, or `http://www.geonames.org/`.

All of the collected services are real, working web services, a significant number of them is commercial. The services have been manually added to OPOSSum with

[24] `http://fusion.cs.uni-jena.de/professur/jgd`

complete information, including natural language documentation of what the service delivers and pointers to the implementation of the services as well as the website from which the service was retrieved. We found that the characteristics of these real services actually differ significantly from the ones of services found in previous test collections. The average number of service input parameters, for instance, is roughly twice as high as in the case of OWLS-TC or SWS-TC (2.6 versus 1.3 and 1.4). This may be due to the chosen domain, but it certainly illustrates the need for versatile data.

The input and output parameter types of the services have been manually linked to WordNet synsets. The natural language documentation of the services was retrieved from the WSDL documentation elements (if available) and the websites of the service providers. In some cases, the services have been invoked and the (sometimes skinny) documentation from the provider websites has been extended based on the gained insights.

All services have been manually tagged with tags linked to WordNet synsets. The tags currently being used are: address lookup, addresses, airport, altitude, articles, bearing, cash machine, city, congressional district number, converter, country, currency, demography, destination point, directions, distance, elevation, geocoding, geographic area, geographic information, ip address, iso code, map, mid point, postal code, public buildings, reverse geocoding, saltwater, search, sunrise, sunset, time zone and weather.

119 of the JGD services are WSDL-based, the others are REST-based services. For the WSDL-based services the original WSDLs are attached to the service entries. If an original WSDL contained several operations, we added those operations that represent a cohesive functionality as a single service to OPOSSum, i.e. we split services if they contained several independent service operations. Thus, original WSDLs attached to a service may describe several more operations in addition to the operation that represents the service they are attached to. Also, different OPOSSum services representing different functionality but resulting from one WSDL will have that same identical (original) WSDL attached.

For better usability, we additionally created derived versions of the original WSDLs by removing the additional operations, bindings, messages and types. Furthermore, we created fictitious WSDL descriptions for the REST-based services that originally did not have WSDL descriptions. Thus, all JGD services have appropriate WSDL descriptions which make them easily usable for WSDL based service tools as well as tools working with the SAWSDL standard. The non-original WSDL descriptions, i.e., the down stripped derived WSDLs as well as those created from scratch for the REST-based services, are clearly marked in OPOSSum by corresponding comments.

The JGD also defines ten sample requests and extensive redundant graded reference relevance judgments for these requests by at least three different relevance

	OWLS-TC	SWS-TC	SAWSDL-TC	Koblenz	ASSAM	DIANE	WS Challenge	SWS Challenge	JGD
Large size	✓	(✓)	✓	–	(✓)	(✓)	✓	–	(✓)
By different groups	(✓)	–	–	–	–	–	–	✓	✓
Different domains	✓	✓	✓	(✓)	✓	(✓)	–	(✓)	–
Realistic services	(✓)	✓	(✓)	(✓)	✓	(✓)	–	✓	✓
Variety	–	–	–	–	–	(✓)	–	(✓)	(✓)
NL descriptions	(✓)	(✓)	–	–	–	✓	–	✓	✓
WSDLs	–	–	✓	–	✓	–	✓	✓	✓
Original real WSDLs	–	–	–	–	✓	–	–	–	✓
Number of formalisms	2	1	2	1	1	1	1	?	4
Documentation	–	–	–	–	(✓)	(✓)	(✓)	(✓)	✓
Intended tests	(✓)	–	(✓)	–	–	(✓)	✓	✓	✓

Table 5.3.: JGD in comparison to previous test collections

judges. Extensive information about these judgments and a service retrieval evaluation experiment that they have been designed for is available online[25]. Furthermore, semantic descriptions in several formalisms for the services from the JGD were contributed by different groups as part of the execution of the mentioned experiment. All these aspects are covered in detail in Chapter 7.

Table 5.3 shows a comparison of the Jena Geography Dataset to the other collections. As is obvious from the table, it does not resolve all issues yet, but significantly improves in the following important aspects:

- It contains contributions by many different groups.
- It is comprised of real, existing services.
- It provides natural language descriptions, WSDLs, and the original WSDL files from which the services were derived.
- It is much better documented than previous collections and comprises detailed specifications about the experiment that it has been designed for.
- Most importantly, it provides alternative descriptions in several formalisms.

[25] `http://fusion.cs.uni-jena.de/professur/jgdeval/`

The primary drawback of the JGD is its limited size. Furthermore, it covers only one domain. While it represents the variety of services within this domain very well and thus, overall, shows a relatively higher variety of characteristics than many existing collections, results obtained using this collection may differ from results obtained with services from other domains. A particularly important characteristic of the JGD with this respect is that the contained services are exclusively data services. They provide or manipulate data, but they do not cause real world effects that involve a lasting commitment of some kind (like the reservation of a flight or the purchasing of an article). It has to be assumed that this may affect evaluation results.

On the other hand, while the collection is comparatively small, it is still the largest collection with real, handpicked services from a single domain and (apart from the OWLS-TC / SAWSDL-TC pair) the only one for which descriptions in different formalisms are available. Furthermore, the JGD was primarily created to perform a retrieval precision experiment for service matchmakers, see Chapter 7. In this context, it was more important to cover one domain comprehensively, i.e. have a lot of related and similar services, than to cover a broad variety of domains. At the same time, compromises with respect to the size of the JGD had to be made. In fact, the 200 services of the JGD were actually already much more than most people participating in the mentioned experiment were willing or able to deal with (see Section 7.8.1).

In summary, the JGD presents an important contribution but obviously not a final solution. In the future, it should be complemented by similar collections from other domains showing different service characteristics (e.g., services that cause lasting real-world effects).

CHAPTER 6

Benchmarking the Functional Scope of SWS Discovery Frameworks

> Measure what is measurable, and
> make measurable what is not so.
>
> *(Galileo Galilei)*

After having discussed issues around test data for SWS evaluation in the previous chapter, this and the following chapter present the two concrete benchmarks contributed within this thesis. This chapter presents a benchmark for evaluating the functional scope of SWS discovery frameworks. This benchmark has been developed under the umbrella of the SWS Challenge initiative. The following Section 6.1 will clarify this relationship. Afterwards, Section 6.2 will define the scope and evaluation goals of the benchmark. Section 6.3 presents the evaluation methodology, i.e., its measures and measurement procedures on a rather abstract level. Section 6.4 presents the concrete benchmark test data and Section 6.5 the corresponding concrete measures. Section 6.6 presents the results from the benchmark and Section 8.6.4 discusses lessons learned during its implementation over the last years. Finally, Section 6.7 concludes the chapter.

The work in this chapter with respect to the benchmarking methodology has been partially published in [PKMS08, PMK+07, LKP+08, PKM+08]. The participation of the author in the concrete benchmarking activity is covered by [KKRK06b, KKR06a, KKR07c, KKR07a, KKR07b, KKR08c] and comparisons between the author's and other people's approaches to problems defined by the benchmark have been published in [KTZ+07, KTKR+08, KKRMS08].

6.1. Relationship to the SWS Challenge Initiative

This benchmark and the underlying benchmarking methodology has been developed within the SWS Challenge initiative introduced in Section 3.1.1, thus some clarifications of the relationship between the benchmark and the SWS Challenge initiative are necessary.

The SWS Challenge is a community initiative dedicated to the collaborative and comparative evaluation of SWS technologies. It is broadly organized into a mediation and a discovery track with different focuses and also slightly different evaluation methodologies (cf. Section 3.1.1). The author has been primarily involved in the discovery track. Therefore, the benchmarking methodology presented in this chapter covers this track, but not the mediation track. Consequently, in the context of this chapter, the term *SWS Challenge* refers to the discovery track of the SWS Challenge and will be used to refer to the initiative representing the reference implementation of the presented benchmarking methodology.

The presented evaluation approach has been developed during the course of various workshops that the SWS Challenge has held since 2006[1], as part of a book project about the SWS Challenge [PLZM08] and within a W3C Incubator Activity[2] [PKMS08]. Therefore, apart from the author, several people have contributed to the benchmark, most notably Holger Lausen (at that time working at STI Innsbruck, Austria), Charles Petrie (Stanford University, USA), Tiziana Margaria (University Potsdam, Germany) and Birgitta König-Ries (Friedrich-Schiller-University Jena, Germany). During the cooperation within the SWS Challenge, the contributions by these and other people were sometimes inseparably mingled. Therefore, the claim here cannot be that every aspect of the benchmark presented below is a personal achievement of the author. During 2006 and 2007, the author has participated in the benchmark. Starting from 2007, he has led the development and execution of the benchmark as part of his position as Discovery Scenario Chair of the SWS Challenge for several years. Where possible, the origins of the presented benchmark will be clarified during its presentation.

6.2. Evaluation Purpose and Scope

The primary evaluation goal of the benchmark is to assess the functional scope and capabilities of frameworks intended to facilitate the support or automation of mediation, choreography and discovery for Web Services using semantic annotations. With this respect, the benchmark aims at providing means for a certification that offers an independent verification that claimed technologies actually work. An

[1] `http://sws-challenge.org/wiki/index.php/Workshops`
[2] `http://www.w3.org/2005/Incubator/swsc/`

intended side-effect of the benchmark is to explore the trade-offs among existing approaches and to figure out which parts of problem space may not yet be covered.

The approach is to define a set of problem scenarios to be solved by participants in an evaluation. The solutions to the scenarios are then used to assess the functional capabilities of participating approaches. This way, the benchmark provides a common ground for comparison among different technologies. The opportunity to compare different solutions to common problems promotes a much deeper understanding of the pros and cons of various technologies and the tradeoffs amongst them than would be possible otherwise. Thus, the comparison enhances the scientific understanding by exploring the fundamental problems and challenges of the research area.

The benchmark aims at not limiting or presupposing the technologies being used to solve a problem in any way. It strives to be open to all kinds of technologies, semantic as well as more traditional ones, able to solve at least parts of the problem scenarios forming the benchmark.

The evaluation approach tests whether a particular problem could be solved by a particular approach or formalism correctly or not. In doing so, the intention is to focus on the *how*, that is the concrete techniques and descriptions an approach uses to solve a problem and not, for instance, on the time it requires for execution. This methodology is described in detail in the following section.

6.3. Evaluation Methodology

The current evaluation methodology has evolved over several years. Some aspects of the original methodology have proven infeasible and are no longer pursued. The corresponding lessons learned will be discussed in Section 8.6.4. Here, only the currently employed measures will be presented. This will be complemented by the description of the procedural setup that has proven successful to collect those measures in the second part of this section.

6.3.1. Evaluation Measures

As mentioned, the measurement approach of the benchmark is to define a set of realistic problem scenarios to be solved by participants. In order to be as objective and unbiased as possible, problem scenarios are not specified in some formal language, but described using English natural language text as well as standardized and well-established technologies like WSDL and XSD. This ensures that as many technologies as possible are applicable to the scenarios and that the scenario specifications provide a level playing ground and are not biased against or in favor of some approaches. The pros and cons of using natural language for problem specifications will be discussed in Section 8.6.4.

To also minimize the potential bias in the selection of scenarios, a formal scenario proposal process is in place within the SWS-Challenge. Scenario proposals need to be submitted to a SWS Challenge workshop where they are discussed in public, usually resulting in clarifications of the scenario semantics and often change requests regarding their design. This process is repeated until a scenario proposal is accepted in consensus by the SWS Challenge steering committee and at least one SWS Challenge workshop and only then becomes part of the official SWS Challenge testbed.

Encouraging the community as a whole to contribute new scenarios to the official SWS Challenge testbed ensures that, over time, this testbed reflects a variety of different perspectives, assumptions and focuses. On the other hand, the consensus decision about the acceptance of new scenarios by the SWS Challenge steering committee and at least one SWS Challenge workshop ensures that only problem scenarios that are sufficiently mature, well specified, relevant and realistic become part of the testbed. Alltogether this contributes to the testbed's and thus the benchmark's quality.

Problem scenarios are typically layered into levels of subproblems which focus on different aspects of the overall problem. This allows a fine-grained evaluation. Usually, problem scenario levels are organized such that the first problem is very basic and subsequent problems add additional complexity on top of the previous problems or focus on complementary but more advanced problem aspects.

As could be expected, the execution of the benchmark within the SWS Challenge showed that different problem aspects are challenging for different approaches. However, it also turned out that approaches often face difficulties that are not anticipated, neither by the scenario designers nor the participants. This reflects the relatively low level of scientific experience and engineering knowledge about the whole problem space and further motivates this particular benchmarking approach.

The critical point of the benchmark is how a certain completeness and relevance of the employed problem scenarios can be promoted and how the certification by means of concrete use cases can be abstracted to more generally valid principles. The approach to the first issue is the above mentioned scenario contribution process that leverages the wisdom of the community in assembling a complete and relevant set of problems.

The approach to the second issue is to examine and compare the various solutions to the concrete scenarios. Such critical examination allows abstracting a list of functional challenges which are valid and interesting beyond the concrete scenarios or solutions. This way, the benchmark methodology allows exploring and delimiting the underlying problem space of interest. Obviously, this is an incremental process as the problem scenarios, the technologies used to solve them and the understanding of the problem space as such co-evolve. The current map of functional challenge will be described in Section 6.5.

The concrete measure employed by the benchmark to assess the functional capabilities of participating approaches is to simply certify the set of concrete problem levels solved by each participating approach. A matrix links the solved problem levels to the functional challenges involved, thus providing an indication about the functional capabilities of the participating technologies. This is a somewhat modest or even humble approach. However, more ambitious measures have been tried, but all been judged infeasible and discarded. This will be discussed in Section 8.6.4.

Instead of just certifying the concrete solved problem levels, one would probably also prefer directly certifying the ability of a technology to solve a particular functional challenge. However, this requires the identification of sets of problem levels covering each functional challenge completely and exhaustively. Unfortunately, it turned out that in correspondance with the relative novelty of the field, the identification of a set of sufficient problem levels associated with a functional challenge is not yet feasible, since unexpected problems and challenges are still identified and the technologies and the requirements to the technologies are still evolving too much.

The current scenarios test, for instance, the capability of frameworks to automatically and autonomously invoke a service (see Section 6.5.8). Thus, it might be assumed that a correct solution of the corresponding goals would be sufficient to certify this ability. However, all of the current goals are based on invocations of SOAP based web services. The ability to invoke REST based services is not yet covered. Similarly, it is unclear whether all challenges potentially involved in the necessary data mediation during service invocation are already covered by the existing scenarios. While this is a rather obvious example, less obvious ones have also occurred (see Section 6.5.9). One has to assume that more will be discovered over time. Only once the field has evolved further and the space of functional challenges has been fully explored and become sufficiently stable, the identification of a necessary and sufficient set of concrete problems associated with each functional challenge will become feasible.

6.3.2. Evaluation Procedures

We now continue with a description of the procedures developed to perform the above described certification. The SWS Challenge offers a continuous call for participation. Potential participants can access the detailed scenario descriptions on a public wiki and start working anytime on any problem of their choice using a technology of their choice.

The scenarios include specifications on what constitutes a correct solution to a scenario and how to verify the correctness of a solution. The concrete layout of these specifications depend on the concrete scenario at hand and will be described in Section 6.4. Typically, participants develop and test their solution until it correctly solves a problem. This is very different from the evaluation approach described in

the following Chapter 7. After developing a solution, a paper describing the solution needs to be submitted to a SWS Challenge workshop. After having passed a regular reviewing process by members of the SWS Challenge program committee that are familiar with the problem scenarios and the benchmarking procedures, the solution needs to be demonstrated at the workshop. By consensus, the workshop verifies whether a problem scenario (or a subset of its problem levels) was correctly solved according to the criteria defined in the scenario specifications.

Typically, but not necessarily, scenarios are accompanied by a testbed implementation, i.e., a set of web service implementations with which solutions need to properly interact in order to solve the problem scenario. For these scenarios, the verification of solutions includes a verification by the benchmark organizers whether the exchange of messages between the scenario solution and the corresponding scenario testbed implementation is correct. This process may be partially automated.

As mentioned previously, the benchmarking methodology does not limit the way how problems are solved or the technologies being used for this. It is open for conventional programming approaches just as well as for semantic frameworks making heavy use of sophisticated reasoning techniques. However, the workshops do verify whether a problem was solved in the way claimed in the solution description. During the workshops, a live code review is performed where participants are asked to explain their solution on the technical code level and may also be required to adapt their solution live to small changes in the problem specifications. This ensures that, for instance, a solution claiming to perform automated service discovery did not hardwire the known correct solution for the corresponding test problems. While tending to be time-consuming, one of the highly positive side-effects of this code reviewing is a much more intense technical discussion leading to a much deeper understanding for each others technologies than could be achieved without the public code review.

After the workshop, a matrix with the evaluation results, i.e. the list of correctly solved problem levels, is published. This list may be complemented by footnotes further illustrating characteristics or restrictions of the certified solutions that the certifying workshop agreed upon.

An explicit goal of the benchmark is to not only certify the capabilities of SWS technologies, but also evolve an understanding of the various technologies and encourage the reuse of them, thus building towards "best practices" whereever possible. Participants are thus generally encouraged to upload their solution to a public FTP server and document it publicly. This allows and encourages other people using the solution to learn about a technology and furthermore promotes independent repeatability and verification of the evaluation results. However, in order to avoid increasing the entry barrier for participants the solution submission and documentation process is not mandatory. Unfortunately, this has resulted in not all current solutions being uploaded or documented properly in the described way.

Having discussed the general methodology of the benchmark, we now turn to describing its concrete implementation, i.e., the scenarios that so far constitute the benchmark problems.

6.4. Problem Scenarios

As of 2009, benchmark comprises six scenarios[3], roughly divided into three mediation and three discovery scenarios. The former focus on service composition and aligning process choreographies whereas the latter are centered around service discovery. Please note that the discovery scenarios do also involve data mediation and, albeit less complex, service composition challenges. In this aspect the mediation and discovery tracks are not entirely disjoint.

The mediation scenarios have been developed by staff from STI Innsbruck and Stanford University and are not within the primary focus of this thesis. Of the three discovery scenarios, the *Shipment Discovery Scenario* has originally been developed by STI Innsbruck and then refined by the author, the *Discovery II and Simple Composition Scenario* has been developed by the author and the *Logistics Management Scenario* has been developed by CEFRIEL, Milano, and the University of Bicocca, Milano in cooperation with the author. Each scenario is described in turn. The functional challenges involved in the scenarios will be discussed in detail in the subsequent Section 6.5.

6.4.1. Shipment Discovery Scenario

The *Shipment Discovery Scenario*[4] poses the problem of dynamically selecting and invoking a shipment service suitable to perform a specific, given shipment request [LKP+08]. The scenario defines five shipping services (described via their WSDL descriptions and additional natural language documentations) that have been modeled after real shipping services and are characterized by the following properties:

- *Operation range:* Shippers operate worldwide or in a set of listed countries or continents.
- *Package limitations:* Shippers define maximum bounds on the dimensions and the weight of packages. Additionally the notion of a *dimensional weight* is used by some shippers: for packages with a low weight, but a large size the dimensional weight (computed from the dimensions of the package) may need to be used instead of the actual weight.

[3] `http://sws-challenge.org/wiki/index.php/Scenarios`
[4] `http://sws-challenge.org/wiki/index.php/Scenario:_Shipment_Discovery`

- *Price:* Four shippers statically specify the price as rules how to compute the price of a package depending on the destination of the shipment and the package dimensions and weight. One shipper requires to dynamically call a Web Service endpoint to gather the current price providing the same information.
- *Package collection:* Shippers offer collection of packages and optionally allow specifying a collection interval during the ordering of a shipment. They define various constraints on the minimum or maximum advance notice for collection or the total length of the collection interval.
- *Shipping time:* Shippers specify rules about the maximum shipping time depending on the destination of the shipment and the time of the collection.
- *Web service interface:* Shippers offer different interfaces to order shipments. During invocation, this requires the lowering and lifting of data to the XML schema of the chosen shipper. Furthermore, some but not all of the shippers support the ordering of multiple packages in a single order. In case of one service, ordering multiple shipments in one order is less expensive than issueing several orders.

Furthermore, the scenario defines nineteen concrete shipping requests grouped into five problem levels:

- *Discovery based on destination:* Two requests are characterized by each defining a specific package (dimensions and weight) that needs to be sent to a given location. Packages dimensions and weights are chosen such that all shippers can handle them, however, not all shippers offer service to the requested delivery addresses.
- *Discovery based on weight:* The three requests defined on this problem level are chosen such that all shippers service the destination address, but not all shippers support the shipment of the packages because of their size and weight. The goals also check the correct implementation of the dimensional weight rules including the correct rounding of dimensional weights.
- *Discovery based on destination and price:* The four requests defined on this problem level specify constraints on the maximum price of the shipment and thus require the implementation of the price rules of the shippers. As mentioned, one shipper requires to dynamically obtain the price of a shipment from a web service endpoint.
- *Discovery for multiple packages:* The five requests defined on this problem level require the sending of multiple packages. Depending on the interface

offered by the chosen shipper (chosen regarding weight, price and destination restrictions), such orders need to be correctly mapped to multiple or single invocations of the corresponding web service.

- *Discovery based on destination and temporal reasoning:* The five requests defined on this problem level specify the current time and add requirements on the collection interval or the maximum shipping time. Thus, solutions need to reason about the current time, the minimum advance notice of the shipping services and the rules about the maximum shipping time in order to identify the matching services. Some goals specify a concrete collection interval, but other goals require to autonomously figure out a collection interval that meets the constraints of the requester as well as the service provider.

To further illustrate the shipping services, we present the details of one shipper and a sample goal.

Racer: The rates are composed of a flat fee and a fee per pound different for every continent: Europe($41.00/$6.75), Asia($47.50/$7.15), North America($26.25/$4.15), rates for South America like North America, rates for Oceania like Asia. Furthermore for each collection order $12.50 are added, regardless of the number of packages collected. Racer ships to 46 countries which are listed in its interface specification (WSDL file). The maximum package weight is 70 lbs. Racer requires at least a pick-up interval of 120 minutes for collection and the latest possible collection time is 8 pm. If a package is collected by 6 pm, it is shipped in 2 business days within a country and 3 business days internationally.

Example Goal E1: One package with dimensions 10/2/3 (l/w/h in inch) weighing 5 pounds shall be shipped from an address in California to an address in New York. The current time is 6:00 AM, the package needs to be collected prior to 9:00 AM and the package has to be delivered at the next business day.

It can be seen that Racer does not qualify for this goal since it does not meet the requirements on the shipping time (three other service providers qualify for this goal).

6.4.2. Hardware Purchasing Scenario

In the second discovery scenario[5], a customer wants to buy computer hardware with fairly clear requirements on the products to buy. Some examples will be provided below. Three services that sell products (called Bargainer, Hawker and Rummage) are defined. Each of the services offers an endpoint that allows to inquire about

[5] `http://sws-challenge.org/wiki/index.php/Hardware_Purchasing_Scenario`

the products (and their detailed properties) currently on stock. Like in the first scenario, the task is to select the right service and invoke it with the right input parameters to purchase the products that best match the customer's expectations. The Hardware Purchasing Scenario was designed to extend the Shipment Discovery Scenario along three dimensions of difficulty:

- Currently, the available services offer a total of 19 products which are identified by a global product id (GTIN) and also fully listed in the scenario description. Clearly, more realistic services offer way more different products. It may or may not be feasible to specify all different options and all the product details in the offer descriptions. Solutions to the scenario should indicate how they attempt to address this issue in more realistic scenarios with thousands of products available. One approach might be to inquire about available products dynamically [KKR07d].
- Some requests contain competing preferences as is usual for realistic matchmaking: price should be as low as possible, processor power, hard disk drive size and memory size should be as big as possible, etc. The scenario request definitions clearly define rankings among such competing preferences. The semantic task is to represent these ranking rules clearly and execute them.
- The scenario requests involve basic service composition challenges:
  1. Unrelated composition: Some requests ask for several products that may or may not need to be purchased from different providers. Thus, a single request needs to be mapped to multiple invocations of the same or different services.
  2. Correlated composition: Some requests ask for several products but not all possible pairings of requested products are feasible. Products may be incompatible to each other or there may be global conditions on the whole set of products to purchase, for instance, on the combined price. Thus, making a choice for one product may limit the choices for the remaining products to purchase or even make it impossible to fullfil the goal.

To further illustrate the scenario, two exemplary goals are provided.

Goal B2: Purchase a 13 inch Apple MacBook with a 2.0 GHz Intel Core Duo processor. It should have at least 1 GB RAM and at least a 100 GB HDD. The price should be around $1500, at the very most $1800. If the white version is significantly cheaper than the black one (at least $100) buy the white one, otherwise buy the black version.

The resulting preferred solution is a white MacBook for $1449 by Bargainer. Another solution, albeit less preferred, is a black MacBook for $1699 by Rummage.

Goal C4: Purchase a 13 inch Apple MacBook with at least 2.0 GHz Intel Duo Core Processor, 512 MB RAM and 80 GB HDD. Additionally buy a web cam for notebooks with a resolution of at least VGA (640*480) and a 13 inch notebook sleeve. The total price must not exceed $1750. As long as the price limit is satisfied, choose the better product: The processor power of the notebook is most important to me. Besides, I rather need more RAM than a bigger HDD. If possible prefer webcams with a higher resolution.

The resulting solutions are as follows: The MacBook can be purchased from Hawker or Bargainer (preferred since better product). The products offered by Rummage either lack processor power or are too expensive after the web cam is added. The web cam needs to be purchased from Rummage since other web cam offers either do not specify a resolution or the specified resolution is too low. Hawker is the only service that offers sleeves.

6.4.3. Logistics Management Scenario

The third and latest discovery scenario[6] is also the most complex one [CCC+08]. It extends and complements the previous scenarios along two dimensions, namely ranking discovered services on the basis of a set of soft contraints and resolving heterogeneity between the provider and the requester perspectives and terminologies.

With respect to ranking and selection the Logistics Scenario complements the Hardware Purchasing Scenario in introducing further problems which require dealing with customer preferences to suitably rank a set of services that all meet the hard constraints of the requester. User preferences need to be expressed in the requests and matched against the given service descriptions. As with the previous scenario, this problem is far from trivial when it comes to expressing priorities among different preferences or choosing optimal compromises in cases of contradicting preferences, for instance, considering price versus quality aspects.

With respect to mediating terminologies, providers and customers in the scenario use different terminologies, because they have different points of view. Hiding this heterogeneity in a mediation system and allowing each of the partners to use the terminology he is familiar with is particularly desirable, if the rules that allow linking a term of one terminology to a term (or structure of terms) of another terminology are very complex. This is the case in the logistics domain covered by this scenario. Complex legal regulations need to be considered that a customer may not know of and does not want to deal with.

[6] `http://sws-challenge.org/wiki/index.php/Scenario:_Logistics_Management`

More concrete, freight can be perishable or dangerous. During the transport of perishable goods certain temperature ranges need to be maintained at all time. The classes of perishable goods, the temperature ranges to maintain for such goods and various types of vehicles able to maintain certain temperature ranges are specified in the international Accord Transport Perishable (A.T.P.) normative[7].

Dangerous goods, on the other hand, include gases, flammable or explosive products, toxic substances, radioactive materials and such. The Accord européen relatif au transport international des marchandises Dangereuses par Route (A.D.R.)[8] regulates the transportation of dangerous goods. It defines nine classes of peril and specifies the constraints that a truck has to meet in order to be admissable for tansporting dangerous goods of the different types.

In the scenario, logistics operators offer transportation of freight between locations and storage capabilities in warehouses. They specify their storage and transport capabilities in terms of the A.T.P. and A.D.R. classes that their vehicles and warehouses support whereas the clients specify the concrete goods to be transported. It is the responsibility of the mediation system to connect these perspectives by reasoning about the applicable A.T.P. and A.D.R. regulations to determine whether a logistics operator is suitable for a given transportation request.

The seven logistics operators that the scenario defines are further characterized via the following properties:

- *Geographic Scope:* They provide transportation within a specified list of countries and continents.
- *Operating Hours:* They offer pickup and delivery of goods within specified daily operating hours.
- *Order Management Speed:* They require a certain time for order handling and management. The time necessary for a transport is the combination of the time necessary for order management and the mere driving time.
- *Prices and payment:* The cost of a transportation is given as a function of a base price and a weight and distance dependent price. Some operators offer discounts if several shipments are ordered. Furthermore, logistics operators offer different payment methods where either the sender ("carriage paid") or the recipient ("carriage forward") pays the freight. Furthermore, payments need to be made within a specified payment deadline.
- *Insurance:* The operators also offer different insurance models where the freight is insured against loss ("refund for loss") or damage ("refund for damage") or both.

[7] `http://www.unece.org/trans/main/wp11/wp11fdoc/ATP-2007e.pdf`
[8] `http://www.unece.org/trans/danger/publi/adr/adr2007/07ContentsE.html`

- *Fleet:* Each operator specifies the list of vehicle types they possess. Vehicles are characterized by the A.T.P. and A.D.R. classes they support as well as their average speed.

- *Warehouses:* Operators may provide storage capabilities in warehouses. Warehouses are characterized by their locations and the A.T.P. and A.D.R. classes they support. Temporary storage in a warehouse either in the pickup or delivery city is necessary if the interval between requested pickup and delivery time exceeds the necessary transportation time by more than twentyfour hours.

To further illustrate the scenario, an exemplary provider and an exemplary goal are provided.

Fresh 'n' Fast Service: It operates in Spain, France, Italy and Germany. Its operating hours are from 4:00 AM till 7:00 PM and it requires 8 hours for order management. The base price is EUR 120, additionally EUR 15 per kg and EUR 0.20 per km are billed. The payment deadline is sixty days from ordering. The only supported billing model is "carriage paid". The only available insurance policy is "refund for loss". Fresh 'n' Fast operates warehouses in Cannes (A.T.P. class "FNB") and Paris (A.T.P. classes "FNC" and "RRC"). It operates with a fleet of pickup trucks (47.5 km/h, A.T.P. class "RRA"), refrigerator trucks (42.0 km/h, A.T.P. class "RNA") and big trucks (35.0 km/h, A.D.R. class 1).

Goal E1: Shipping of fruit ice cream to be picked up in Milano, Italy on 11/09/2008 10:00 (GMT+1) and delivered in Paris, France on 27/09/2008 14:30 (GMT+1). A total of seven shipments is requested. Ideally, the base price of the shipment should be less than EUR 250. The payment deadline should be between 45 and 60 days. Insurance, both for loss and damage is preferred.

As can be seen, this goal requires temporary storage in a warehouse because the pickup and delivery date are far apart. Furthermore, A.D.R. normatives need to be checked for both the vehicle and the warehouse since ice cream requires cooling. The goal specifies three distinct preferences on price, payment deadline and insurance. This does not establish a total ordering among the alternative services. Web service 1 (Liteworld), for instance, is preferable to Web service 7 (GTL) with respect to all three preferences. However, Liteworld is preferable to Fresh 'n' Fast with respect to price and insurance, but less preferable with respect to the payment deadline. The goal does not specify clear criteria how to deal with this situation, thus, different rankings are equally acceptable.

6.5. Functional Challenges

After having introduced the problem scenarios, we now turn to discussing the functional challenges involved in these scenarios. Some of the challenges were explicitly designed during the development of the problem scenarios. However, during the development of solutions to the scenarios based upon different SWS approaches unexpected challenges or challenge aspects were discovered. The current list of challenges was assembled by analyzing and comparing the various SWS Challenge solutions. Basically, every problem characteristic that was perceived as requiring certain distinct capabilitites within the solution frameworks or that posed particular difficulties to at least one framework was associated with a, potentially new, functional challenge.

This implies that the identification of functional challenges is an iterative process resulting in the continuous refinement of the challenges list. It is thus not claimed that the current list is complete. In fact, the identification of new challenges is expected to continue with the definition of new problem scenarios or new attempts to solve the existing scenarios with other SWS frameworks. Nevertheless the current list at least represents a proven framework for comparison of the existing solutions to the current problem scenarios. The functional challenges have been grouped into the following categories:

1. Basic discrete matchmaking
2. Matchmaking with numbers
3. Matchmaking with temporal reasoning
4. Rules
5. Preferences, ranking and selection
6. Composition
7. Mediation
8. Advanced matchmaking aspects.

The functional challenges of each category will be introduced and illustrated by examples from the scenarios in turn. Table 6.1 shows the complete list of functional challenges and the goals from the scenarios associated with each challenge. Please note that full information about the corresponding goals is available online[9].

[9] `http://sws-challenge.org/wiki/index.php/Scenarios`

Functional Challenge	Related Goals from Scenarios		
	Shipment	Hardware	Logistics
1. Basic discrete matchmaking			
Discrete conditions	A1, A2, C1–E5	A1–C4	A1–E1
Hierarchical concept inclusion	A1, A2, C1–E5		A1–E1
2. Matchmaking with numbers			
Numeric comparisons	B1–C4, D2–D5	A1–C4	A1–E1
Arithmetic computations	B2–C4, D2, D4, D5	B2, C2–C4	D1, E1
3. Matchmaking with temporal reasoning			
Comparison of time instances	E1–E5		A1–E1
Special time notions	E1–E5		
Computations with time	E1, E2, E4, E5		A1–E1
Computations with special time notions	E4, E5		
4. Rules			
Conditional expressions	E3–E5		
Conditional matchmaking rules	C1–C4, D2, D4, D5		A2–E1
5. Preferences, ranking and selection			
Discrete preferences		B2	B1, C1, E1
Continuous preferences		B1, B2, C2, C4	C1, D1
Multiple prioritized criteria		B2, C4	C1
Multiple unprioritized criteria			E1
Relative preferences		B2	
6. Composition			
Unrelated composition	D1, D3	C1	
Correlated composition	D2, D4, D5	C2–C4	
7. Mediation			
Data mediation between the syntactic and the semantic level	A1–E5	A1–C4	
Data mediation on the semantic level			A1–E1
Process mediation	D2, D4, D5		
8. Advanced matchmaking aspects			
Uncertain matchmaking results due to lack of information			D1
Inherently uncertain results	E5		
Performing service calls	A1–E5	A1–C4	
Dynamic information	C1–C4, D2, D4, D5	A1–C4	
Domain functions			A1–E1

Table 6.1.: Overview of functional challenges

6.5.1. Basic Discrete Matchmaking

The most basic functional challenge refers to checking basic discrete conditions. These come in two flavors.

Discrete conditions check (a finite list of) single, discrete requirements on, for instance, the color of a product to be purchased, the processor type of a notebook, the country of a destination address, etc. Different requirements may be alternative or conjunctive.

Hierarchical concept inclusion requires to reason about sub- and supertypes or other forms of concept inclusions, for instance, for determining whether an address which only specifies a country is located in a certain continent. Implementation alternatives include object-oriented inheritance, logic subsumption reasoning or rules.

6.5.2. Matchmaking with Numbers

Most scenarios require the ability to process numbers. This is certainly not trivial in all cases. The necessary introduction of so called *concrete domains* in description logics most often used in the Semantic Web, for instance, may have drastic effects on the computational complexity of these logics and is thus not supported in all description formalisms [Lut02]. Matchmaking with numbers requires two basic capabilities.

Numeric comparisons refers to the ability of processing numbers with respect to relations like *equal*, *smaller*, *greater*, etc. This is required, for instance, to check limitations on the weight of a parcel or the price of a product.

Arithmetic computations refers to the capability of computing functions on numbers like the basic *sum*, *product*, *division*, etc. or more complex ones like the trigonometric functions. This is required, for instance, to compute the dimensional weight of a package based on its dimensions, the combined price of a set of products to purchase or the price difference between alternative products.

6.5.3. Matchmaking with Temporal Reasoning

Handling temporal aspects requires to reason with numbers, but additionally demands the handling of time-specific objects like days, months, business days, time zones etc. More precisely, the following four challenges can be distinguished.

Comparison of time instances refers to the ability to compare time and date instances. This is, for instance, necessary to compare the pickup and delivery

times of a package with the business hours of a shipper or to check whether the estimated shipping time of a parcel meets the constraints of a requestor. Note that this, for instance, may also require to compare concrete and specific points in time, with general reoccuring time intervals like "between 10 AM and 8 PM".

Special time notions refer to concepts that go beyond simple, numeric, time and date values. This includes relative time notions like *now*, *today*, *tomorrow*, *this week*, calendar related aspects like *holidays* and *business days*, *time zones* etc. Shippers, for instance, may offer pickup and delivery of packages only on business days. The computation of the earliest possible pickup time requires a notion of the current time, i.e., *now*. Similarly, constraints on the shipping time may refer to concepts like *tomorrow*.

Arithmetic computations with time require to compute time intervals and durations based upon time and date instances. This ability is needed, for instance, to compute the delivery time of a parcel based upon the time of pickup and the estimated shipping time or to compute the earliest possible pickup time based upon the current time and the minimum advance notification interval.

Arithmetic computations with special time notions combine the requirement to perform temporal computations with the usage of special time notions, for instance, to express limitations like *within three business days*.

6.5.4. Rules

The ability to express rules beyond basic discrete matchmaking rules is central to many use cases. Such rules can be broadly classified in two types.

Conditional expressions are needed when different values must be used for the evaluation of some matchmaking rules depending on other property conditions. The price formula to be used to compute the price of a shipment in the Shipment Discovery Scenario, for instance, depends on the destination continent and must be chosen properly when checking price limitations of a shipping request. Similarly, the shipping time specified in numbers of business days depends upon the time of pickup (prior or after a specified time of the day).

Conditional matchmaking rules are needed when some matchmaking restrictions need to be checked only conditionally. For instance, a warehouse is needed in the Logistics Scenario if and only if the time between pickup and delivery of the parcel exceeds the shipping and handling time for more than 24 hours. A.T.P. and A.D.R. capabilities of trucks and warehouses need to be considered only, if the freight has certain properties.

6.5.5. Preferences, Ranking and Selection

Preferences are used to rank a list of matches in order to select the best one. This either refers to a situation with approximate, i.e., imperfect matches, or to a setting where a request explicitly distinguishes between hard constraints that must be fulfilled by any candidate service and soft constraints (often called non-functional properties or preferences) that are desireable but not mandatory and can thus be used to order the list of candidate services that match the hard constraints of the requestor. The handling of preferences and ranking involves the following challenges.

Discrete preferences establish an ordering based upon a finite number of discrete preference levels. A request might specify preferences on insurance of parcel from "no insurance" (least preferred), over "insurance for damage" and "insurance for loss" to "insurance for damage and loss" (most preferred). Similarly, a requestor might specify that he prefers black notebooks over white ones. Discrete preferences typically correspond to basic discrete matchmaking conditions.

Continuous preferences introduce a notion of "better fulfillment" over an infinite basic set. Usually, this requires to handle numbers and corresponds to matchmaking with numbers. Goals in the Hardware Purchasing Scenario, for instance, prefer products with lower prices or higher capabilities (hard disc space, memory size, processor power, etc.)

Multiple prioritized ordering criteria involve multiple ordering criteria (either discrete or continuous) that are clearly prioritized. Thus, an umambiguous ranking is established even in the presence of conflicting optimization goals. Goal C4 of the Hardware Purchasing Scenario, for instance, specifies that the price limit is most important, but as long as that is satisfied, better products should be preferred. With respect to better products, the processor power of the notebook to be purchased is most important and more RAM is considered more important than a bigger HDD.

Multiple unprioritized ordering criteria refers to a situation with multiple ordering criteria that are not clearly prioritized. I.e., the specified preferences establish a partial instead of a full order among the candidate services. This is relevant since often it may not be feasible to resolve conflicting optimization goals in a satisfying way without complete knowledge about the landscape of available offers. The matchmaking system will have to handle such situations either by offering alternative rankings, partial rankings (i.e. groups of comparable services), or by autonomously serializing the partially ordered results into a completely ordered result list. Goal E1 of the Logistics Scenario, for instance, prefers shippers with a base price of less than EUR 250, a payment deadline

between 45 and 60 days and insurance for both, loss and damage, but does not specify how to rank shippers that fulfill different subsets of these preferences.

Relative preferences specify preferences not on an abolute level, but relative to alternative candidate offers. A requester might, for instance, prefer a given service he has good experience with, unless another service is cheaper by at least 20%. Similarly, Goal B2 of the Hardware Purchasing Scenario prefers a black notebook, unless the white version is significantly (at least $ 100) cheaper than the black one. While absolute preferences allow assigning a preference value to each offer to later sort the offers based on their preference values, relative preferences require the direct comparison of alternative services in order to rank them properly. In the examples above, for instance, preference values for the services can not be assigned independently from the alternative services.

6.5.6. Composition

Service composition and discovery are closely related tasks. On the one hand, discovery precedes composition if the component services to be composed are not known in advance (this is not the case with the SWS Challenge mediation scenarios which focus on the mediation and planning aspects involved in service composition). On the other hand, service composition is necessary during service discovery, if no single service is able to fulfill a request, but multiple services are able to deliver the required functionality if they are properly combined. With respect to such situations, it is clearly advantageous to integrate composition capabilities into service discovery frameworks [KKRKS07]. Therefore, the presented benchmark contains basic composition challenges which can be broadly categorized in two types.

Unrelated composition allows splitting a request into multiple requests that are not related to each other and can be served independently. I.e., unrelated composition challenges can be reduced to a sequence of single requests. In the Shipping Discovery Scenario, for instance, several requests requires the shipping of multiple packages. Only in the presence of price limitations and for a single service (Racer) it matters whether these multiple packages are treated as several independent shipping requests or are treated as a single order. In the Hardware Purchasing Scenario, some goals require the purchase of multiple products from the same or different vendors without linking the different products to each other using some global requirements.

Correlated composition extends the previous case such that the multiple invocations can not be handled independently of each other. Therefore, the matchmaking needs to be aware of composing several services to fulfill a global

request. The purchase of a notebook and a compatible docking station in the Hardware Purchasing Scenario, for instance, requires to consider compatibility of notebooks and docking stations. Other goals in that scenario enforce global conditions on the overal purchase, for instance, a limit on the total price of the package. In such cases, the choice of one service may limit the choices for other parts of the request or even render the total request infeasible. Techniques like backtracking or multi constraint optimization are necessary to correctly handle this challenge. In the Shipment Discovery Scenario, the price of the shipment of multiple packages via Racer is less expensive if they are requested in a single order and thus only require a single package collection. This needs to be considered when different services are combined for sending different packages, potentially leading to quite complex optimization and planning problems.

6.5.7. Mediation

Mediation refers to the necessity of resolving heterogeneities and incompatibilities. These may occur in the data representation, or in the process choreographies of the involved services.

Data mediation between the syntactic and the semantic level is also called lowering/lifting. It refers to the process of translating data between the semantic, ontological representation (typically used within the formal reasoning engine of a discovery framework) to a syntactic representation of that data, like the XML messages commonly consumed and produced by web services. The complexity of such translations may differ widely, from simple differences in the serialization of strings (e.g., names, addresses, country names, etc.) to significant structural differences in the representation of the data (e.g., from a set of RDF triples to a tree based structure like XML). Lowering and lifting is required for all goals within the Shipping Discovery Scenario and the Hardware Purchasing Scenario at least in order to invoke the selected service.

Data mediation on the semantic level refers to the necessity to mediate different data representations or terminologies on the semantic level. In the case of the Logistics Scenario, the terminologies of the provider, who specify their capabilities in terms of A.D.R. and A.T.P. regulations, and the requester, who specify their requirements in terms of concrete goods, need to be reconciled. Other, less complex examples involve the conversion of data values specified in different units. Note that this may require obtaining dynamic conversion rates in case of currencies.

Process mediation is required by the Shipment Discovery Scenario where the interface of some services offer the ordering of multiple packages but the interface of other services only offers the ordering of a single package. If multiple shipments are requested, the meditor needs to reason about the interface of the chosen shipper and either map a single goal to several service calls, or to a suitable package order. Note that process mediation may involve data mediation. The given example, for instance, requires to create different XML messages to properly represent a combined package order or several distinct service calls.

6.5.8. Advanced Matchmaking Aspects

Advanced matchmaking aspects cover several challenges not falling in any of the previous categories. These include matchmaking under uncertainty, performing and evaluating service calls, retrieving and leveraging dynamic information as well as representing domain functions or relations.

Uncertainty arises if the available information during the matchmaking is insufficient to certainly determine whether or how well a service matches a request. This may be caused by a failure of one of the parties (service requestor or provider) to specify the necessary information, but it may also be that the information is simply not available at the time of the matchmaking.

Uncertain matchmaking results due to lack of information refer to a situation where the matchmaking results are uncertain because one of the available parties (requester or provider) has failed to specify a necessary piece of information. In Goal D1 of the Logistics Scenario, for instance, one of the services offers a price discount if multiple shipments are ordered. However, this policy was not known to the requester beforehand, thus, the requester did not specify the number of shipments to be ordered. A discovery framework may resolve this situation by either presenting a conditional result list or by checking back with the user to inquire about the number of shipments.

Inherently uncertain matchmaking results are a special case of the previous one. Goal E3 of the Shipment Discovery Scenario, for instance, requires next day delivery of the shipment and the current time is 10:00 AM. The only candidate service requires a collection interval of five hours and no advance notification. Therefore, collection will happen between 10:00 AM and 3:00 PM. However, the service guarantees next day delivery only if the parcel is collected before 2:00 PM. Thus, in contrast to the other services, this service offers a fair chance to meet the constraint on the shipping time, but there is no way to guarantee the match. The service can be considered a potential match.

Performing and evaluating service calls tests the basic ability of frameworks to automatically invoke a discovered service. This requires the lowering and lifting of data (see above) and the proper implementation and execution of the corresponding protocols, like SOAP. The Shipment Discovery Scenario as well as the Hardware Purchasing Scenario require the automatic invocation of the most suitable service at the end of the discovery process. The calling of remote services may result in numerous errors. A service may have become unavailable, the network may fail, the data lowering and lifting process may fail or the remote service may locally throw an exception. Various strategies may be employed in the case of failed invokations, for instance, the invocation of another candidate if the first one failed. This may require further action, like compensation of already finished service invocations. This proper reaction to service invocation errors is currently not explicitly covered by corresponding scenario goals.

Dynamic information refers to a situation where some piece of data necessary for the matchmaking can not be encoded statically in the service descriptions, but need to be obtained dynamically by calling a corresponding service enpoint. This is the case in the Shipment Discovery Scenario where one of the services does not publish a price policy but offers a service endpoint to inquire about the price of a given package. Similarly, the services of the Hardware Purchasing Scenario offer a dynamic listing of the available products. Matchmaking with such dynamic information requires invoking services during the matchmaking process (including the associated challenges), processing the service responses to make the contained information available to the matchmaking and (possibly iteratively) continue the matchmaking with the newly obtained information.

Representation of domain functions relates to the encoding and representation of functions and relations specific to certain domains and too complex or data intensive to be statically encoded in a suitable domain ontology. The Logistics Scenario, for instance, requires to determine the distance between the source and destination of a shipment to compute the shipping time that is necessary to determine whether a service is a match and whether a warehouse is necessary. In the scenario, the occuring destinations are statically provided and can thus be statically encoded. However, a more realistic scenario would encode and obtain this information otherwise. A possible solution is to treat this as dynamic information and call a web service endpoint to obtain the necessary information. However, for performance reasons other solutions may be preferable. The corresponding challenge is currently not explicitly covered by any scenario goal.

6.5.9. Conclusions

The identification of functional challenges from concrete problem scenarios is not always easy, in particular, since some approaches face problems with aspects that other approaches can easily handle [KTKR+08, KKRMS08]. Moreover, problems can not always be foreseen and are sometimes even unintuitive.

The Jena solution, for instance, despite of being among the most complete certified solutions to the SWS Challenge discovery scenarios, was unable to solve a certain goal of the Hardware Purchasing Scenario. This goal required to reason about compatibility between notebooks and docking stations. This compatibility was represented via a property of each docking station that listed the compatible notebooks. While DSD, the language used by the Jena approach, was capable of expressing that a single valued attribute has to be member of a given set, it was not capable of expressing the reverse case, i.e., that a set based attribute has to contain a given individual [KKR07c]. However, this was necessary to express that the docking station property specifying the list of compatible notebooks must contain the product identifyer of the notebook to be purchased (the chosen notebook is compatible to the chosen docking station). Even though this was a deficiency of the language implementation, rather than a fundamental lack of expressivity, the example illustrates that the identification of suitable tests to reliably certify a functional capability can be quite tricky.

Therefore, and as discussed above, the claim is not that the presented list of functional challenges (and thus the benchmark) is complete. The given list rather represents the current state of development of the benchmark for the functional scope of discovery frameworks. While the benchmark is not complete, and does not claim to be complete, the methodology of

1. developing concrete problem scenarios,
2. assessing their relevance through a community process,
3. abstracting fundamental functional challenges from the concrete problems and the concrete problem solutions and
4. assessing these again through a community process

has proven to be feasible, productive and successful. The repeated assessment through a community process, in particular, ensures a quality management which is otherwise difficult to maintain in the absence of well-established processes and quality criteria. Along with the desired evolution of the benchmark, the presented methodology will continue to identify new challenges until, eventually, the problem space is sufficiently explored and understood and a true standard benchmark has been established.

6.6. Evaluation Results

The discovery problem scenarios were evaluated at the SWS workshops in Budva, Montenegro (2006)[10], Athens, GA, USA (2006)[11], Innsbruck, Austria (2007)[12], Stanford, CA, USA (2007)[13] and Einhoven, The Netherlands (2009)[14].

So far, a total of five teams successfully submitted solutions to the scenarios: A joint team from Politecnico Milano (Italy) and Cefriel (Milano, Italy), another joint team from University Milano-Bicocca (Italy) and Cefriel (Milano, Italy), a team from DERI Galway (Ireland), a team from University Jena (Germany) and a joint team from University Dortmund (Germany) and University Potsdam (Germany). Please note that primarily because of the significant effort involved, not all teams attempted solving all scenarios. Therefore, a solution not certified for a particular problem level does not necessarily imply the inapplicability of the team's technology to the corresponding functional challenge.

Table 6.2 shows an overview of the official evaluation results. The footnotes within the table denote limitations of the solutions that have been agreed upon during the evaluation workshops. Please note that, over time, the existing problem scenarios have evolved. Furthermore, the Logistics scenario was only recently added. Evaluation results are presented in their accumulated version as of 2009. Furthermore, evaluation results will be presented based upon the problem level hierarchy which was valid at the time of the corresponding evaluations. This hierarchy differs slightly from the current one which underlyies the overview of functional challenges from Table 6.1. The corresponding changes will be discussed in Section 8.6.4.

Finally, please note that in addition to the pure evaluation results, in-depth comparisons of different technologies have been performed based upon the solutions to the problem scenarios. These comparisons were jointly written by the authors of the compared solutions [KTZ+07, KTKR+08, KKRMS08, KVW+08]. This project greatly increased the mutual understanding for each other technologies and the tradeoffs involved in them. A summary of these comparisons including a description of the different problem solutions is provided in Appendix B.1.

6.7. Summary

This chapter presented a benchmark for assessing and certifying the functional scope of SWS discovery frameworks. The purpose of this benchmark is threefold:

[10] http://sws-challenge.org/wiki/index.php/Workshop_Budva
[11] http://sws-challenge.org/wiki/index.php/Workshop_Athens
[12] http://sws-challenge.org/wiki/index.php/Workshop_Innsbruck
[13] http://sws-challenge.org/wiki/index.php/Workshop_Stanford_(2)
[14] http://sws-challenge.org/wiki/index.php/Workshop_ECOWS_2009

Problem Level	PC*	BC*	DG*	JE*	DP*
Shipment Discovery Scenario					
2a: Discovery based on Destination	✓		✓	✓	✓[1]
2b: Discovery based on Destination and Weight	✓		✓	✓[2]	✓[1]
2c: Discovery based on Destination, Weight and Price	✓		✓	✓	✓[1]
2d: Discovery involving simple composition	✓			✓	
2e: Discovery including temporal reasoning	✓			✓[3]	
Hardware Purchasing Scenario					
3a: Discovery based on clear defined product specifications — Goal A1	✓		✓	✓	
3a: Discovery based on clear defined product specifications — Goal A2	✓		✓	✓	
3b: Additionally specifying preferences — Goal B1	✓		✓	✓	
3b: Additionally specifying preferences — Goal B2				✓	
3c: Unrelated composition of services — Goal C1	✓		✓	✓	
3c: Correlated composition of services — Goal C2				✓	
3c: Composition of services (unrelated but global condition) — Goal C3			✓	✓	
3c: Composition of services (unrelated with global condition and preferences) — Goal C4			✓	✓	
Logistics Management Scenario					
A1: Standard single order		[4]			
A2: A.D.R. rules		[4]			
A3: A.T.P. truck		[4]			
B1: A2 plus simple soft constraints		[4]			
C1: A3 plus soft constraints with preferences		[4]			
D1: Warehouse		✓[4]			
E1: A.T.P. truck plus warehouse		✓[4]			

*) PC: Politecnico Milano - Cefriel; BC: University Milan Bicocca - Cefriel; DG: DERI Galway; JE: University Jena; DP: University Dortmund - University Potsdam

[1]) No automated invocation
[2]) Arithmetic calculation performed by external Web services (which is absolutely good)
[3]) Algorithm is correct, but not complete
[4]) The representation and execution of the A.T. P. and A.D.R. regulations as well as the preference policies were solved correctly, but there were bugs in the underlying functional discovery with respect to the computation of shipping times and the corresponding filtering of providers.

Table 6.2.: Functional Scope Benchmark results

1. Assessing and independently verifying claimed capabilities of SWS discovery technologies.

2. Learning about these technologies, in particular investigating the tradeoffs of different approaches by comparing their application to a common set of problems.

3. Exploring the general problem space of SWS discovery by identifying a list of fundamental functional challenges in the area.

The first goal is approached by specifying a set of concrete and detailed problem scenarios, layered into different subproblems focusing on different aspects of the overal problem. Interested participants may use their technology to solve the provided problem scenarios and submit a description of their solution to a dedicated workshop series. The solutions are then presented at a workshop, including a live demo and a technical review on the code level. By consensus, the workshop decides whether a problem aspect was solved correctly. An official list with the set of problem aspects certified to have been solved correctly is publicly available on the Web and updated after each workshop. The relevance of the specified problems and thus the certification results is ensured by an acceptance process for new problem scenarios which includes presentation of scenario proposals at at least one evaluation workshop and consensus acceptance of the scenario by the SWS Challenge steering committee and the community as represented by the workshop participants.

The second goal is achieved through the live demo of the solutions at the evaluation workshop. The included technical code review by the workshop participants ensures a detailed insight into the characteristics of the various solutions. Furthermore, participants are encouraged to team up and prepare papers with a detailed comparison and joint analysis of their solutions. These detailed comparisons based upon concrete solutions to common problems further increase the mutual understanding for each others technologies and their pros and cons. Moreover, since the comparisons are jointly written by the developers of the compared technologies, they provide outsiders with the opportunity to learn about their tradeoffs in an objective way.

The third goal is tackled by abstracting fundamental functional challenges from the concrete problem scenarios that have been presented by the community. This abstraction process leverages the experience of the participants regarding the challenges they encountered during the implementation of their solutions. Again, a community reviewing process is used to ensure a proper level of acceptance of the identified challenges in the community. The identification of fundamental challenges underlying concrete problems allows a more systematic approach to the research in the area and promotes a common understanding and vocabulary for the relevant problems.

The current state of the benchmark, its measures, procedures, concrete benchmark problems and evaluation results have been presented. The benchmark should continue to evolve with the scientific progress in the area and the increasing understanding for the general problems involved. Corresponding options for the future evolution of the benchmark will be discussed in Chapter 9. A critical review of the benchmark and a discussion of the lessons learned during the participation and organization of the benchmarking campaign will be provided as part of the validation of this thesis in Section 8.6.4.

CHAPTER 7

Benchmarking SWS Matchmaking

> Good benchmarks are like good laws. They lay the foundation for civilized (fair) competition.
>
> *(Kim Shanley)*

This chapter presents a benchmark for evaluating SWS discovery and matchmaking approaches. The task of comparing semantic goal descriptions with semantic offer descriptions to determine services relevant to a given task is involved in almost all use cases around SWS technology. Therefore, the evaluation of SWS matchmakers is at the very core of SWS technology evaluation. The presented benchmark is primarily concerned with evaluating the retrieval correctness of SWS matchmakers, but also covers runtime performance, usability and coupling characteristics to some extent. The description of the benchmark is complemented by a presentation of background work that the benchmark builds upon and an extensive analysis of the reliability of the benchmarking methodology.

7.1. Chapter Organization

The chapter is organized as follows. Section 7.2 describes the scope and use case of the benchmark. This will be followed by a discussion of the state of the art in the area in Section 7.3. This discussion will reveal several shortcomings and open problems. A new setup for the evaluation of SWS matchmakers which overcomes these problems is presented in Section 7.4. Subsequently, the main building blocks of the benchmarking approach will be covered.

First, the central notion of *relevance* between a service request and a service offer will be discussed. The state of the art will be examined and a novel relevance model that overcomes existing shortcomings will be presented (Section 7.5).

Second, questions regarding the reliability of reference relevance judgments obtained from human assessors will be discussed. The consistency of such judgments will be experimentally investigated and the effects of different relevance models to the consistency of judgments analyzed. Recommendations how to obtain reliable judgments will be derived (Section 7.6).

Third, measures to quantify and compare the retrieval correctness of matchmakers will be covered. Desirable measure characteristics will be defined and measures from information retrieval be discussed with respect to these characteristics. It will be shown that existing measures suffer from problems. Solutions to these problems will be proposed (Section 7.7).

Having discussed the theoretical background and building blocks of the benchmarking approach, JGDEval (Jena Geography Dataset Evaluation), a reference execution of the benchmark, will be presented in Section 7.8. It describes the used dataset, the participating approaches, the evaluation environment and the results of the evaluation.

The results from JGDEval will then be used to investigate and discuss the reliability of the benchmarking approach in Section 7.9. The analysis covers the effects that the choice of relevance, inconsistency in reference relevance judgments and the choice of evaluation measure have on the evaluation results. Recommendations to ensure the maximum reliability of evaluations performed according to the methodology presented in this chapter will be given.

The chapter concludes with an overview of the related work in Section 7.10 and a summary of the contributions of the chapter in Section 7.11. The work presented in this chapter has been partially published in [KKR08a, KKR09].

7.2. Evaluation Purpose and Scope

The presented benchmark targets the use case of a human developer who is searching for a web service that provides functionality needed in some application being developed. Currently, a developer will query and browse a web service registry (like seekda.org, programmableweb.com, or xmethods.com) to identify promising candidate services. Semantic descriptions are expected to make such manual discovery more efficient by improving the filtering and ranking of the services in the registries. It is the aim of this benchmark to test this hypothesis and investigate the strengths and weaknesses of current approaches by comparing the performance of different semantic and non-semantic service retrieval approaches.

The use case underlying this chapter is related to but different from the use case in Chapter 6. The latter defined concrete requests which should result in the automated selection and invocation of a suitable service. The current one does not require automated invocation and thus poses less hard requirements to the correctness of the matchmaking process. Thus, it is more suitable for open environments and allows a broader range of approaches, in particular with respect to differences in how comprehensively service semantics are formalized. Furthermore, the previous use case asked for the delivery of a business service (like the transportation of a concrete package) whereas this one requires the discovery of web services as a means to repeatedly access such business services (like a web service offering transportation services) [Pre04].

The main questions that should be investigated by this evaluation are:

- How precise, complete and efficient are current technologies for service retrieval?
- How much information needs to be shared between providers of the service descriptions and developers posing service queries to allow for efficient retrieval?
- What is the right level of detail to describe services for the given task of retrieval from a registry?
- How is the trade-off between description effort and retrieval precision?
- What is the best pattern to describe services?
- What is the most suitable formalism to do so?
- Which retrieval techniques are good for which retrieval problems? What are the properties that make a specific retrieval problem difficult for some or all techniques? What features of services make their correct and precise retrieval difficult for certain or all approaches?
- Does semantic retrieval indeed improve retrieval precision compared to traditional technologies (structural WSDL matchmaking, natural language processing, keyword-based search, ...)? What is the involved extra cost (e.g., for developing the semantic descriptions) for the improvement?

Note that some of these questions are evaluative and should be answered in a q uantitative way. In particular the retrieval precision achieved by current semantic technologies should become measurable in a quantitative, objective way. However, most questions are rather investigative than evaluative. Determining the properties that make a specific retrieval problem difficult for some or all techniques, for instance, can hardly be achieved by some quantitative measurement alone, but rather

through the in-depth knowledgeable analysis and interpretation of such measurements. Therefore, while the benchmark will provide an answer to questions of the first kind, it can only strive to provide the data that enables experts to investigate questions of the latter type.

7.3. State of the Art

The matchmaking use case described above can be considered as a special instance of an information retrieval problem. Therefore, it is not surprising that similarly targeted SWS matchmaking evaluations have almost exclusively employed the prevailing evaluation methodology from IR, i.e., the Cranfield paradigm as, for instance, represented by the Text REtrieval Conference (TREC)[1].

According to this paradigm, the retrieval efficiency of an IR system is mainly evaluated by means of *recall* and *precision*. Recall is a measure for the completeness of the retrieval and defined as the number of relevant documents retrieved by the system divided by the total number of relevant documents. Precision is a measure for the correctness of the retrieval and defined as the number of relevant documents retrieved by the system divided by the total number of retrieved documents. *Relevance* is typically based on topical similarity and obtained from *reference relevance judgments* by domain experts. Test collections therefore have three components:

- a set of documents (the test data),
- a set of information needs (called topics or queries) and
- a set of relevance judgments (lists of documents which should be retrieved for each query) [Voo01b].

While this evaluation approach has dominated SWS retrieval evaluation so far (see Section 3.1 and 7.10), there are significant differences between traditional information retrieval and SWS retrieval that need to be taken into account.

Any retrieval system attempts to satisfy a real world need based upon a supply of available resources. For the retrieval process, the need and the supply are abstracted to a model that supports the system in determining whether a given resource is relevant for a given query. In the case of Web search engines, for instance, such a model will consist of descriptors extracted from the query string and data structures like indexes built upon descriptors extracted from the Web pages etc. The power of this model (how well it represents the user need and the available supply and how well it supports the supply's retrieval, i.e. the filtering and ranking process) is of critical importance for the retrieval system and thus a central component of its overall performance.

[1] `http://trec.nist.gov/`

Traditional IR systems create the model they operate on in an autonomous fashion. Thus, from the viewpoint of an evaluation they operate on the original data. Consequently, very different IR systems can be evaluated on common test data like a collection of documents.

SWS retrieval follows a different paradigm. Here the model is formed by the formal semantic annotations which are usually not created automatically by the retrieval system, but written manually by human experts explicitly to allow for the precise and efficient retrieval of the resources. Notably, there is not yet an agreement about which formalism and model to use for such semantic annotations. In fact, it is one of the core open research questions in the area which semantic model (like WSMO, OWL-S, SAWSDL, ...) offers the best compromise between usability, expressivity and computational complexity. These considerations have important implication for the evaluation of SWS retrieval engines.

First, evaluating how well services are retrieved based upon their formalized semantics really implies measuring the mixed effects of four different factors:

1. The expressivity of the used formalism (how precisely and comprehensively can the semantics of service offers and requests be formalized).

2. The quality of the annotations (how precisely and comprehensively have the semantics of service offers and requests been formalized).

3. The capabilities of the algorithm that operates on the annotations (how effectively and efficiently can the available data be processed).

4. The alignment of the annotations and the algorithm that processes them (to what extent can the information represented in the annotations be exploited).

In particular the second and fourth factor are typically not an issue in traditional IR evaluation, where the existing data is usually processed automatically.

Second, it is well known that some assumptions underlying the Cranfield paradigm are not strictly valid. Relevant documents are usually not equally desirable but differ in their degree of relevance to a given query. Additionally, the relevance of one document is often not independent of the relevance of other documents and thus the user information need while browsing a result list not static. A single set of judgments for a query is not necessarily representative of a whole user population since different users may have different preferences regarding the relevance of documents. Also, the the list of relevant documents for each topic is not always complete, i.e., not necessarily all relevant documents are known in an evaluation. Finally, reference relevance judgments are generally known to differ across judges and for the same judge at different times. The question whether these issues invalidate evaluations based upon the Cranfield paradigm has been subject of intensive discussion in the IR community over decades [Sar95, SJ95, Voo98, Voo01b].

The main conclusions have been that evaluation scores obtained following this paradigm are valid, but only in comparison to scores computed for other systems using the exact same collection and only, if results are averaged over many queries executed on large test collections [Voo01b].

As discussed in Section 5.2 and in contrast to traditional IR evaluation, input data for SWS retrieval evaluation, i.e. semantically annotated service descriptions, is not readily accessible. Thus, test data for SWS retrieval has to be developed explicitly for evaluation purposes. However, test collections applicable to matchmakers using different formalisms are difficult to create and have also not been available (see Chapter 5). Furthermore, due to the effort inherently involved in manually creating semantic service annotations, SWS test collections will remain significantly smaller than traditional IR test collections for quite some time to come. This raises skepticism about the reliability of IR evaluation methodologies when applied to SWS matchmaking.

All these issues need to be taken into account when designing a well-founded evaluation approach for SWS matchmakers. Current approaches to SWS retrieval evaluation (see Section 3.1) are based on test collections of semantic service descriptions in a specific semantic formalism. This results in several problems.

1. Only matchmakers relying on that specific formalism can participate in a corresponding evaluation which greatly limits the scope and thus, potential impact of the evaluation.

2. Such a setting does not really allow to make the formalism itself subject of the evaluation.

3. Evaluation results may be compromised, if there is a lack of alignment of the modeling approach represented by the provided service descriptions with what the various evaluated matchmakers expect. The SAWSDL standard, for instance, leaves great flexibility in the actual shape of semantic annotations. Thus, current SAWSDL matchmakers make assumptions on how the SAWSDL descriptions are designed [KKZ09]. They will not operate as expected, if the processed SAWSDL descriptions do not conform to these assumptions.

4. Generally, descriptions can be (unintentionally) biased towards certain approaches, in particular, if they origin from a single group.

7.4. A New Setup for the Evaluation of SWS Matchmakers

In this section, a novel approach to SWS matchmaking evaluation that overcomes the limitations discussed above is presented. This approach is based upon a test

collection of services described by exactly the documentation that a human programmer would use when selecting a service, i.e., natural language documentation and, if applicable, a WSDL description of the service interface. A corresponding test collection, the Jena Geography Dataset (JGD), has been presented in Section 5.4.

Basing a test collection on natural language service descriptions as opposed to formal semantic service descriptions avoids any bias that may be introduced by the inevitable abstraction process involved in creating initial formalizations of the data. This way, a level playing field for any participating approach is promoted. Another important advantage is that the necessary human relevance judges may assess a service-query pair exactly as they would if they were actually searching for a service themselves. Thus, it is expected that evaluation results become more reliable compared to the state of the art where relevance judges assess the relevance of a service-query pair based upon – potentially inappropriate or incomplete – semantic formalizations. A welcome additional bonus is that relevance judges do not need to be Semantic Web experts anymore.

These advantages are expected to outweigh problems arising from ambiguities in natural language service descriptions that may affect the appropriateness of formal semantic descriptions and the reliability of the relevance judgments. The issue of ambiguities in natural language service descriptions will be further covered in Section 7.6.

Obviously, in order to perform the actual matchmaking evaluation, semantic descriptions for the services in the test collection are required, ideally in several different formalisms. The evaluation approach requires the prospective participants in an evaluation campaign, preferably the developers of the participating matchmakers, to create the required semantic annotations themselves. This ensures high quality descriptions and more specifically guarantees that the descriptions correspond to the modeling style and design patterns most suited for the given matchmaker. It also avoids that participants may challenge the evaluation results based on critique of the used service descriptions, an unsolved issue with the state of the art. Finally, this approach will naturally result in annotations following different modeling paradigms and styles and also different levels of description comprehensiveness. This will allow making the effects of these variations subject of the evaluation, too.

With prior knowledge of the service offers and requests and given the expected relatively small size of the test collections[2], it is relatively easy to design service descriptions in a way that ensures correct retrieval, e.g., by customizing request formulation to the matching offer descriptions – an issue which has not received sufficient attention so far [KKRPK08]. Thus, the evaluation setup must ensure that

[2]The described approach was implemented based upon the JGD, a collection of 200 services. Unfortunately, this already overcharged participants such that the dataset had to be reduced to 50 services (see Section 7.8).

the service requests are formalized by people different from those, who annotated the service offers, and without knowledge about the available service offers. Ideally, the developers of a participating matchmaker are split into two groups, one that annotates the service offer descriptions and another one that annotates the service request descriptions. Furthermore, all information (ontologies, categorizations, description templates, etc.) that is being passed from the former to the latter group needs to be documented. This mimics real world conditions where the offer descriptions are created by the service providers and, independently thereof, the request descriptions by the developers seeking appropriate services. Furthermore, the documentation of the information which has been passed during the evaluation allows drawing conclusions on the kind and amount of information that service providers and service requesters would have to agree upon and share in real world settings.

The above considerations lead to an experimental schedule with eight tasks. Each task will be defined in the following via its actors, its actor model, its inputs and outputs as well as any constraints on the task. The tasks are not strictly ordered, Figure 7.1 illustrates the dependencies between the different tasks.

Task 1: Collect natural language service offers

Actor Model	None
Actors	Organizers of the benchmarking event
Inputs	None
Outputs	A collection of services described in natural language
Constraints	None

As a first step, the organizers of the benchmarking event need to assemble a collection of real (or realistic) services described in natural language and, if applicable, via WSDL descriptions. Collecting the services in a structured form (e.g., in some database) is usually advantageous for the further processing.

Task 2: Define natural language service queries

Actor Model	Web service consumer
Actors	Organizers of the benchmarking event
Inputs	Services from test collection
Outputs	A collection of realistic natural language service queries
Constraints	None

Complementing the collection of services, a number of service queries need to be defined. Typically, knowledge about the service collection is necessary to define queries to ensure that the test collection contains a sufficient number of relevant

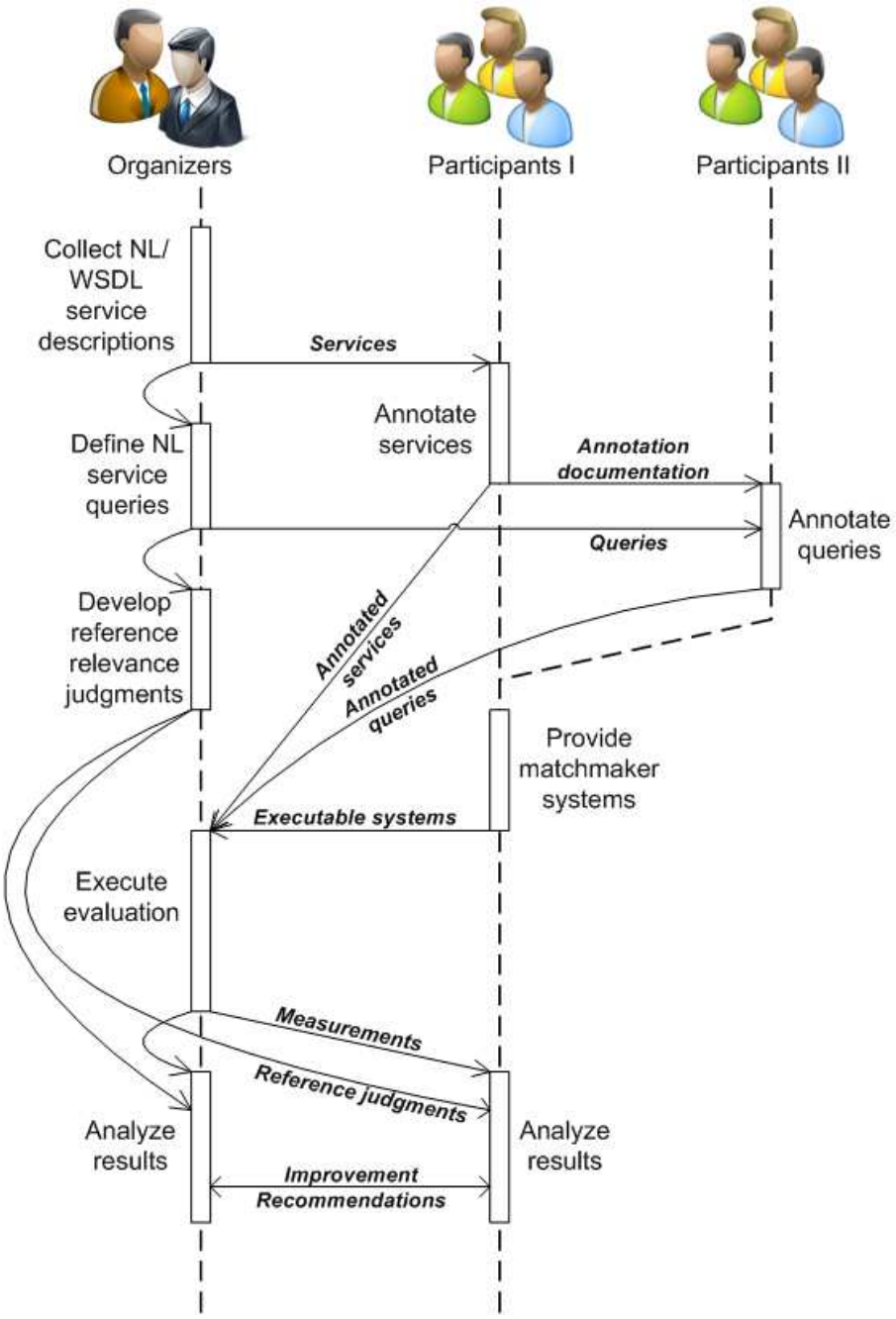

Figure 7.1.: Experimental schedule for the evaluation of SWS Matchmakers

services to each query. It may be useful to encourage participants to suggest queries. This increases the variety of the queries and the acceptance of the evaluation.

Task 3: Develop reference relevance judgments

Actor Model	Web service consumer
Actors	Organizers of the benchmarking event
Inputs	Services and requests from test collection Previously agreed upon relevance judgments guidelines
Outputs	Reference relevance judgments
Constraints	None

Based upon an agreed upon definition of relevance and clearly specified relevance judgments guidelines, the organizers of the benchmarking event need to provide reference relevance judgments that specify the relevance of each service to each query. Aspects around this task will be detailed in Sections 7.5 and 7.6.

Task 4: Annotate services

Actor Model	Web service provider
Actors	Evaluation participants (SWS experts, ideally developers of the evaluated matchmakers)
Inputs	Services from test collection Documentation of discovery engine and description formalism
Outputs	Annotations for services and the necessary resources (ontologies, XML schemas, etc.) Annotation documentation (instructions for service requesters possibly including schemas, ontologies, templates, etc.) Ideally information about the necessary effort involved in creating the annotations
Constraints	Actors must not have access to or knowledge about the queries from the test collection.

Each group participating in the evaluation needs to create and submit annotations for the services in the test collection, including any necessary resources like ontologies or schemas. The group's retrieval system will later use these annotations to retrieve the services. Having each group creating their own set of descriptions as opposed to creating one set of descriptions for each formalism ensures that each matchmaker has a set of optimally suited descriptions. Furthermore, it allows making the description style and the flexibility of matchmakers subject of a cross evaluation if several groups create alternative descriptions using the same formalism.

In order to mimic realistic environments, participants must not have access to or knowledge about the queries from the test collection while creating the service annotations. In addition to the service annotations, annotation documentation for service consumers needs to be assembled during the process of service annotation. The annotation documentation contains all instructions and information necessary for clients to use the group's system to retrieve services. This may include ontologies to be used, vocabularies, request templates, etc. Ideally, participants should report about the effort they invested in creating the necessary annotations.

Task 5: Annotate queries

Actor Model	Web service consumer
Actors	Evaluation participants (SWS experts, ideally developers of the evaluated matchmakers)
Inputs	Queries from test collection Documentation of discovery engine and description formalism Annotation documentation
Outputs	Annotations for queries and updated necessary resources (ontologies, matchmaking rules, etc.) Ideally information about the necessary effort involved in creating the annotations
Constraints	Actors must not have access to or knowledge about the services from the test collection or the service annotations (must be different from actors of Task 4).

Participants need to express the queries of the test collection for usage with their retrieval system. Again, they ideally should report about their effort involved in this task. In order to mimic realistic environments, the people that express the queries must not use any information beyond the provided queries and annotation documentation. In particular, they must not have access to or knowledge about the service offers or the service annotations. Therefore, each participating research group needs to have at least two persons involved in the evaluation. It should be noted though that this requirement can not be strictly enforced since cheating by participants can not be entirely prevented. This should not be critical as long as the primary goal of the evaluation is to learn about each other's approaches.

Task 6: Provide matchmaker implementations

Actor Model	Service registry provider
Actors	Evaluation participants
Inputs	Format specification by organizers of the benchmarking event
Outputs	Executable matchmaker implementations
Constraints	None

Participating groups need to provide an executable implementation of their matchmaker implementation according to specifications by the organizers of the benchmark. Typically, access to the system will have to be standardized in some way, e.g., by implementing a defined interface.

Task 7: Execute evaluation

Actor Model	None
Actors	Organizers of the benchmarking event
Inputs	Service annotations, query annotations and resources created by participants Matchmaker implementations provided by participants
Outputs	Runtime measures Ranked output lists by participating matchmakers for all queries
Constraints	Execution must be performed in a controlled environment and repeated several times.

The organizers of the benchmarking event install the provided matchmaker implementations on a dedicated machine and execute the evaluation. I.e., they feed the service annotations into the systems and query them with the queries as expressed by the participants. This process results in runtime performance measures (execution time, memory consumption) and the ranked output lists of the matchmakers for each query. In order to create reliable runtime performance measures, the execution needs to be performed in a controlled manner and repeated several times.

Task 8: Analyze results

Actor Model	None
Actors	Evaluation participants and organizers of the benchmarking event
Inputs	Runtime measures Ranked output lists Reference relevance judgments from test collection
Outputs	Performance scores Improvement recommendations for matchmakers Improvement recommendations for benchmark
Constraints	None

Together with the participants, the evaluation organizers analyze the measurements collected during the evaluation and compute performance scores for the participating systems. The corresponding performance measures need to be well-defined and specified prior to the evaluation. Based upon the evaluation results and experiences, improvement recommendations for both, the participating matchmakers and the benchmark, should be collected.

7.5. Relevance for SWS Retrieval

After having presented the experimental setup of the benchmark, we now turn to discussing its main components. Since requirements to test data have been discussed in Chapter 5 already, we start here with the reference relevance judgments crucially necessary for any TREC-style retrieval evaluation. Saracevic summarized the relationship between relevance judgments and IR evaluation as follows:

> "As mentioned, IR tests are based on comparing systems relevance – responses to a query that a system deemed and retrieved as relevant following whatever procedure – and user relevance – user's (or a surrogate's) assessment as to relevance of retrieved answers or of any information or information objects in the system, even if not retrieved. User relevance is the gold standard against which system relevance, that is, system performance, is compared. Thus, performance assessment of a given system (algorithm, procedure...) follows from and is based on human judgment of relevance of given information or information object to a given query or need. The key issue is obtaining acceptable relevance judgments that can then be used as a standard for calculating recall and precision. Once these are obtained, calculations are straightforward. Well, almost. [...]

> Establishing this gold standard is one of the main problems, even conundrums, of IR testing. Not surprisingly then, in many reports of IR tests, the critical step showing how relevant objects became relevant is often shrouded in mystery. Or, it is glossed over. Or, it is accepted from a previous source without further ado. Or some collective group, such as 'judges' or 'librarians' or 'searchers' or 'students' is mentioned as bearing the responsibility. Or, some such explanation. It is hard to get at it." [Sar08]

This quote sheds light on the two Achilles heels of relevance judgments used for retrieval evaluations. The first problem is how to properly obtain reliable, high-quality reference judgments from something as unreliable as human assessors. This issue will be dealt with in the following Section 7.6. The other problem to start with is how a notion as fuzzy – and at the same time as important – as relevance can be captured properly in the first place. This will be dealt with now.

7.5.1. State of the Art

Despite the fact that, as Saracevic also remarks, everybody intuitively understands the notion of relevance, "there were, still are, and always will be many problems with relevance. This is not surprising. Relevance is a human – not a system's – notion and human notions are complex, even messy" [Sar07a]. Relevance has been discussed extensively in the IR community (see [Miz97, Sar07a, Sar07b] for overviews) but when it comes to (semantic) service retrieval evaluation, Saracevic's above quoted expression of "shrouded in mystery" describes the situation most properly.

Web Service Retrieval

Dong et al., for instance, have presented a similarity search engine for web services and evaluated their engine based on a test collection of 790 services. They provide an extensive discussion of their evaluation and evaluation results, but provide no information about the definition of relevance applied beyond "similar operations" having been judged as relevant [DHM+04].

Stroulia and Wang [SW05] as well as Kokash et al. [KvdHD06b] present comparable WSDL matchmakers and evaluations based upon a dataset originally created by Kushmerick and Hess[3]. Stroulia and Wang present an extensive discussion of the performance of their algorithm, but do not provide any information about the underlying relevance or relevance judgments at all. Kokash et al. specify that they classified their test services into categories or groups and treated services in

[3] Available at `http://www.andreas-hess.info/projects/annotator/`

one category as relevant to each other. However, no further information about the classification of the services is provided.

Semantic Service Retrieval

Tsetsos et al. were the first to discuss issues around relevance in the context of SWS matchmakers [TAH06]. They discussed deficiencies of a binary relevance approach and presented a graded relevance scale. They evaluated their approach based upon graded relevance judgments for a subset of OWLS-TC. The scale they employed defined five relevance values: *Irrelevant*, *Slightly relevant*, *Somewhat relevant*, *Relevant*, *Very relevant* but no further definitions or information is given.

OWLS-TC is the most widely used test collection in the area of SWS retrieval and most SWS matchmaker evaluations that we are aware of are in some way based upon it, e.g., [KFS06, KK07, KB08]. With respect to its reference relevance judgments, the different versions of OWLS-TC provided the following information and definitions (quoted from the manuals available from SemWebCentral[4]).

The OWLS-TC 1.0 and 2.0 (2005), 2.1 (2006) and 2.2 (2007) manuals state:

> "set of [...] services that we subjectively defined as relevant according to the standard TREC working definition of binary relevance (`http://trec.nist.gov/data/reljudge_eng.html`): *'Only binary judgments (relevant or not relevant) are made, and a document is judged relevant if any piece of it is relevant (regardless of how small the piece is in relation to the rest of the document).'* "

Obviously, this is a rather unspecific definition of relevance.

The very recent OWLS-TC 3.0 manual (2009) presents improvements:

> "The graded relevance sets have been created using the following 4-graded scale:
>
> - highly relevant (value: 3) - Any service offer that is exactly what the user asked for (or even better for him, e.g. by giving additional information)
> - relevant (value: 2) - Any service offer that might answer the request completely or does the requested job at least partially
> - potentially relevant (value: 1) - Any service offer that may be helpful.
> - nonrelevant (value: 0) - Anything totally irrelevant to the service request

[4] `http://projects.semwebcentral.org/projects/owls-tc/`

> [For the binary case] the collaboratively created graded relevance assessments were projected onto a binary relevance scale using SWSRAT. The approach of relaxed binary relevance is chosen for this, which means that a service offer is considered as binary relevant to a query, if it is at least potentially relevant according to the graded scale given above."

While this is a much more specific definition than that from previous versions, concrete judgment instructions, illustrating examples and further information about the process that was used to obtain the judgments are still lacking.

Conclusions

While the survey above is not exhaustive, it gives an appropriate illustration of the state of the art with respect to relevance and relevance judgments in the area of (semantic) service retrieval evaluation. It should be noted that the given examples, while being in need of improvement, still stand out of the large body of similar work without any experimental evaluation at all. Nevertheless they illustrate that the crucial issue of relevance clearly received insufficient attention in the field. In the following sections, we will present our approach to a well-specified relevance model for service retrieval.

7.5.2. A Novel Relevance Model for (Semantic) Service Retrieval

In the area of service retrieval, with the exception of the most recent edition of the S3 Contest in October 2009, all matchmaker evaluations have so far exclusively relied on binary relevance. However, most SWS matchmakers support various levels of match, i.e. establish the relevance based on a graded scale. In a classic paper, Paolucci et al., for instance, proposed the use of *exact*, *plug in*, *subsumes*, and *fail* [PKPS02]. This scale or variations thereof have been adopted by many approaches. Some matchmakers, in particular hybrid ones, establish relevance even on a continuous scale.

The use of binary relevance for SWS discovery evaluation has thus been criticized as too simple and thus not suited to differentiate sufficiently precisely among SWS matchmakers [TAH06]. It is thus desirable to employ a graded relevance scale instead of a binary one in SWS retrieval evaluations. However, the design of such a scale is far from trivial.

To be practically useful it must have clear definitions that enable domain experts to provide reference relevance judgments as unambiguously as possible. In this aspect a scale like *very relevant*, *relevant*, *somewhat relevant*, *slightly relevant*, and *irrelevant* as used by [TAH06] is very difficult to judge objectively. On the other hand, a relevance scale should not depend on a particular matchmaking or formalization approach. It is therefore not appropriate to directly use the degrees

of match by Paolucci et al. as a relevance scale for general SWS retrieval evaluation either.

A general relevance scale used in the evaluation of different SWS retrieval algorithms must meet two basic requirements:

1. To be equally applicable to different approaches and avoid any bias, it must be defined on the level of the problem to solve and not in terms of a technology that is a candidate to solve the problem.

2. To be practically useful it must have clear definitions that enable domain experts to provide reference relevance judgments as unambiguously as possible and allow a third party to understand and properly interpret these judgments independently.

In previous work, we proposed a graded one-dimensional relevance scale for service retrieval evaluation that is based on a set-theoretic service matchmaking model proposed by Keller et al. [KKR08a, KLL+05]. We also performed a preliminary retrieval evaluation experiment based upon this relevance scale. We found that there was significant inconsistency among different relevance judges [KKR08a].

This motivated us to investigate the judgment behavior of the different judges in detail to track down the specific service characteristics that caused human assessors to judge a service-request pair the way they did. By analyzing the services that each judge had assessed a certain relevance level, we identified different complementary notions of relevance that influenced the judges' decisions. This led to the definition of the following multi-dimensional graded relevance model.

Multi-Dimensional Graded Relevance

The multi-dimensional relevance scale distinguishes three different aspects of relevance:

- *Equivalence* determines the functional equivalence of the offer and the request. Does the offer provide qualitatively exactly the desired functionality or only something similar? Possible values are *Equal*, *PossEqual*, *Approximate*, *PossApproximate*, and *Not Related*.

- *Scope* determines the functional completeness of the offer with respect to the request. Does the offer provide quantitatively all the functionality that is requested or just parts? Possible values are *ExcessMatch*, *Match*, *PossMatch*, *Partial*, *PossPartial*, and *NoMatch*.

- *Interface* determines whether the offered interface matches the requested one. Are all offer inputs available in the expected format and does the offer provide

all requested outputs in the expected format? Possible values are *Compatible*, *PossCompatible*, and *Incompatible*.

Please note that, while the three aspects are separated, they are not completely independent from each other. In particular, we decided that if *Scope* or *Equivalence* were judged as *Not Related* respectively *NoMatch* the service should be judged as completely irrelevant (*Not Related*, *NoMatch*, *Incompatible*).

Please also note that service descriptions (regardless of whether formulated in natural language or a formal formalism) will often be incomplete and thus leave room for assumptions or interpretations. The relevance levels starting with *Poss* address this notion of incomplete information which is present in all three dimensions. A service judged *PossEqual* on the Equivalence dimension, for instance, is judged to be certainly *Approximate* and possibly even *Equal*. However, the available information is insufficient to decide this with certainty.

Finally, the *ExcessMatch* on the *Scope* dimension has been defined to characterize a situation where a service provides more than is requested by the client and the additional effects need to be considered potentially harmful and unwanted. Whether additional effects need to be considered harmful or not depends on the use case at hand. Typically, in case of data services, additional information that can be easily filtered by the client will not be considered harmful. On the other hand, in case of services that are not web-safe[5], additional effects will more often be considered harmful. A client that seeks to purchase a cell phone, for instance, may or may not be willing to purchase a phone that comes bundled with a plan of a particular provider.

To further illustrate the above given relevance definitions, assume a request for a US geocoding service, i.e. a service that provides the geographic location of a given unstructured US input address.

- A service that offers geocoding in the UK is considered completely irrelevant to this request although the functionalities are similar and the interfaces might match syntactically.

- A service that provides geocoding of US cities would be judged *Approximate* on the Equivalence dimension (since geocoding on the city level only roughly approximates geocoding on an address level), *Match* on the scope dimension and *Compatible* on the Interface dimension if the service is able to extract the city from the unstructured information, *Incompatible* otherwise or *PossCompatible* if the service documentation doesn't state this information.

[5]Property of an interaction which does not have any significance of taking an action other than retrieval of information, according to the W3C Web Services Glossary at `http://www.w3.org/TR/ws-gloss/`

- A service that provides geocoding of Californian addresses would be judged *Equal*, *Partial*, *Compatible* because it provides the desired functionality, but only for a subset of the potential input space (only Californian addresses).
- A service that provides geocoding of structured addresses only would be judged *Equal*, *Match*, *Incompatible*, since the functionality matches but the interfaces are incompatible.

One-Dimensional Graded Relevance

In order to investigate the effects of multi-dimensional relevance and compare it to other relevance models, we also defined a one-dimensional graded relevance scale which is a slightly disambiguated version of the one we proposed in [KKR08a] and that originally motivated the definition of the multi-dimensional relevance. The scale defines the following relevance levels:

- *Match:* The offer matches perfectly with the request.
- *PossMatch:* The offer might match perfectly with the request. The available documentation is insufficient to tell with certainty.
- *ParMatch:* The offer provides parts of the requested functionality.
- *PossParMatch:* The offer might provide parts of the requested functionality.
- *ExcessMatch:* The offer provides the requested functionality but additionally would result in undesirable effects that are not requested by the client and should be considered harmful and unwanted.
- *RelationMatch:* The offer provides a functionality that is qualitatively similar to the requested one (i.e. the requested functionality could be approximated with this offer) or the offer provides the desired functionality, but the interfaces do not match.
- *NoMatch:* None of the above, the offer is irrelevant to the request.

The first five levels should be applied when the interface of the service possibly matches the desired one and the service possibly provides the requested functionality. The differentiation corresponds primarily to the Scope dimension of the multidimensional relevance scale. If uncertainty is involved in any dimension, the corresponding *Poss-* value should be used. Services that only possibly approximate the desired functionality or offer an incompatible interface are covered by *RelationMatch*. Other services are considered irrelevant. Therefore, the examples listed

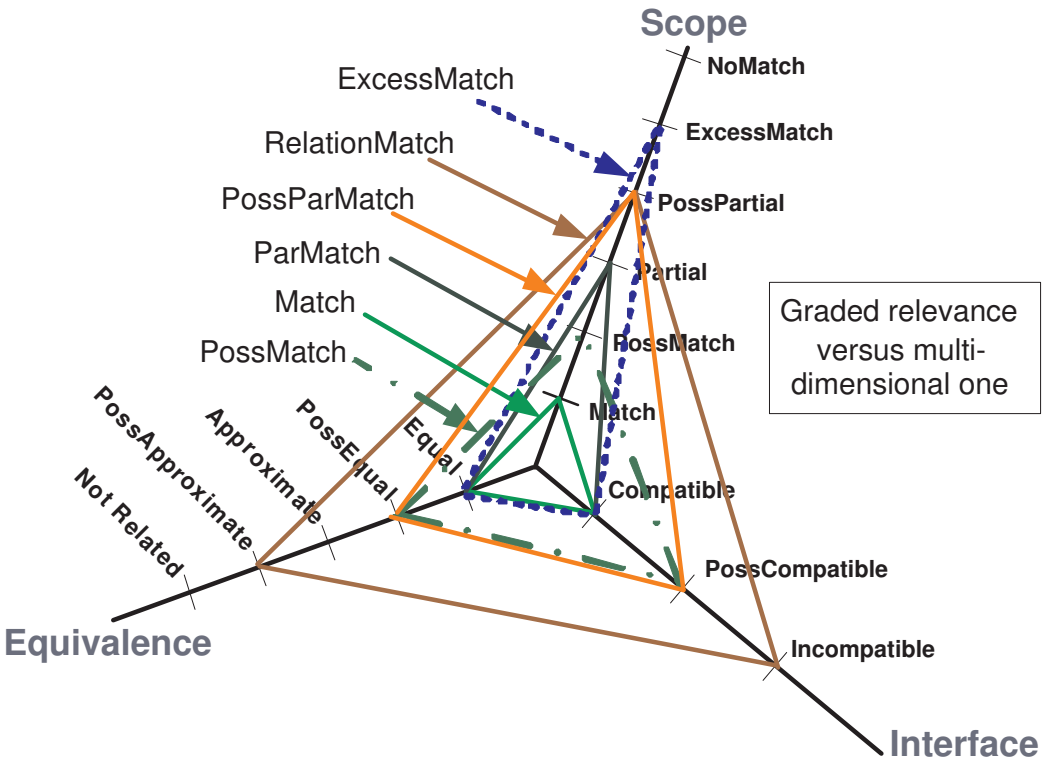

Figure 7.2.: Relationship between the one-dimensional graded relevance scale and the multi-dimensional one

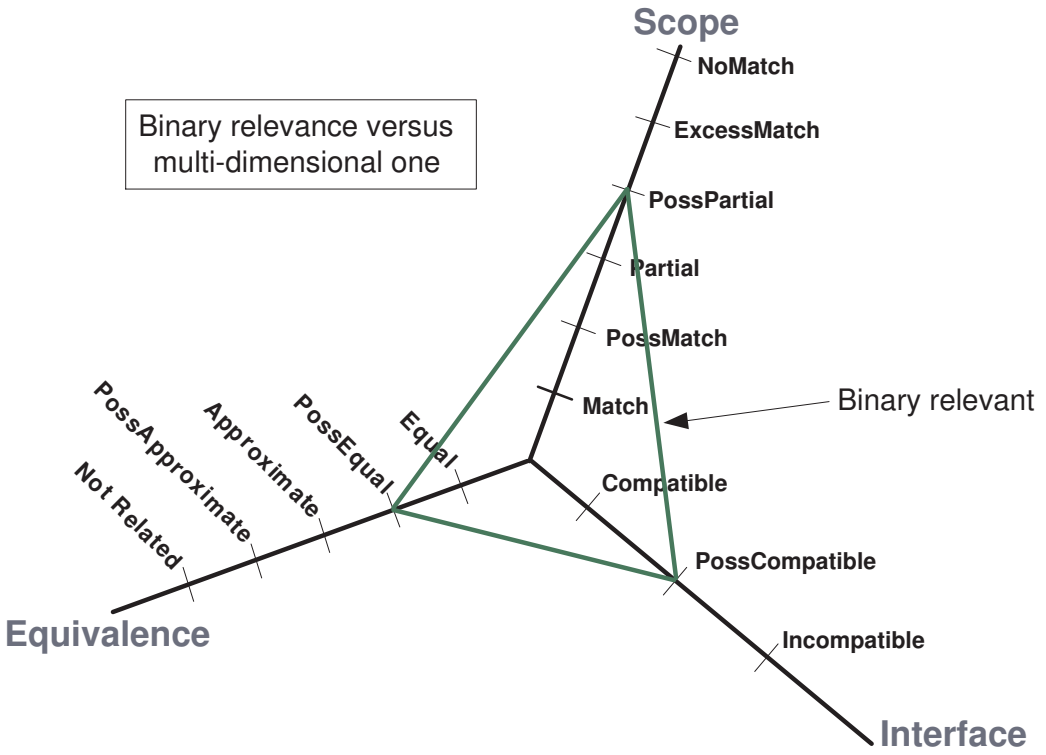

Figure 7.3.: Relationship between the binary relevance scale and the multi-dimensional one

above would be judged *NoMatch*, *RelationMatch*, *ParMatch* and *RelationMatch* respectively. Figure 7.2 shows the precise relationship between the one-dimensional graded relevance scale and the multi-dimensional one.

Binary Relevance

Finally, we provide a binary relevance definition that will be used for further comparison of the two alternative scales provided above. For this scale, we defined *RelationMatch*, *ExcessMatch* and *NoMatch* of the one-dimensional graded relevance scale as *Irrelevant* and the other four levels as *Relevant*. Figure 7.3 shows the relationship between the binary relevance scale and the multi-dimensional one.

7.6. Reliability of Reference Judgments

After having defined a well-founded relevance model for SWS retrieval, we now turn to the second problem of how to obtain reliable reference relevance judgments. There is ample evidence from the IR community that human assessors providing the reference relevance judgments for retrieval test collections disagree substantially. Voorhees remarks with this respect:

> "Inconsistency – the fact that different relevance assessors produce different relevance sets for the same topics – has been the main perceived problem with test collections since the initial Cranfield experiments [...]. The main gist of the critics' complaint is that relevance is inherently subjective. Relevance judgments are known to differ across judges and for the same judge at different times [Voo01b].

While this issue, to the best of our knowledge, has never been discussed in the area of SWS retrieval, it has been a subject of a few dedicated studies in the IR community. Saracevic has recently published comprehensive overviews of the corresponding literature and provides some "hypothetical generalizations" with respect to studies about the question of "how consistent, or rather how inconsistent are relevance judgments?"

> "The inter- and intra-consistency or overlap in relevance judgments varies widely from population to population and even from experiment to experiment, making generalizations particularly difficult and tentative.
>
> - However, it seems that higher expertise and laboratory conditions can produce an overlap in judgments up to 80% or even more. The intersection is large.
> - With lower expertise the overlap drops dramatically. The intersection is small.
> - In general, it seems that the overlap using different populations hovers around 30 percent.
> - Higher expertise results in a larger overlap. Lower expertise results in smaller overlap.
> - Whatever the overlap between two judges, when a third judge is added it falls, and with each addition of a judge it starts falling dramatically. Each addition of a judge or a group of judges reduces the intersection dramatically. [...]" [Sar08]

In the context of SWS retrieval evaluation, this raises the question of whether the presented findings apply here, too. Or, more precisely, given that web service retrieval is a much more restricted domain and use case than general IR, do relevance judgments for service retrieval test collections differ as much as is known from IR? Furthermore, how do different models for relevance, as presented above, influence the consistency of reference judgments? Finally, is it possible to overcome disagreement and develop reliable, consistent reference judgments in the area of SWS retrieval?

In the following, the setup and results from an experiment designed to answer these questions will be presented. The results will show that relevance judgments in the area of SWS retrieval are significantly inconsistent, but that much more consistent judgments can be obtained by using redundant judgments and additional conflict reconciliation effort. Section 7.9.2 will then discuss the effects of inconsistent relevance judgments to the reliability of retrieval evaluation experiments.

7.6.1. Experimental Setup

Ten fictitious service requests were formulated with respect to the Jena Geography Dataset. Out of these, three were selected such that they were different in nature but all had a large number of services from JGD matching to various degrees, thus yielding sufficiently rich data for comparing reference judgments.

The first request asked for a service that converts US addresses to their geographic locations (*US Geocoding*), the second one asked for a service that provides the distance (straight line or driving distance) between two cities world-wide (*Location Distance*), and the third request asked for a service that provides as much information on a given US city-state combination as possible, with zip code(s), area code(s), and the geographic location or area being required for a perfect match (*US City Data*).

Four relevance judges (two semantic web services experts and two computer science students with extensive programming experience) were given detailed instructions to the three relevance scales. A Web portal was provided that allows selecting a request and sequentially the service offers, displays the selected pair with all available information next to each other and supports to conveniently input relevance judgments using a standard HTML form.

Each relevance judge got his own version of the Web portal to prevent them from seeing the judgments by the other judges. All judges judged the complete set of offers with respect to all three requests and the graded and multi-dimensional relevance scales. The 201 services from the Jena Geography Dataset were randomly divided into two groups and the judges judged the services with respect to the different relevance scales in different sequences to avoid an influence of the order in which the services were judged or the relevance scales were used. However, the binary judgments were only later added for comparison, i.e. they were also com-

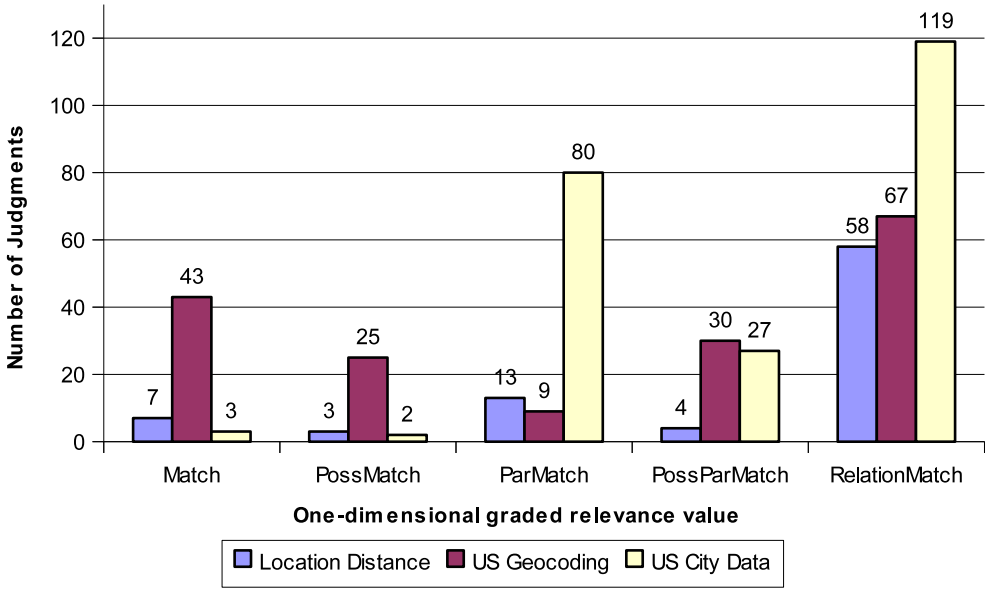

Figure 7.4.: Distribution of one-dimensional relevance values by request

pletely and independently provided by the four judges but only after the judges had completed the judging according to the other relevance scales. Overall, 12060 judgments, 2412 binary as well as one-dimensional judgments (4 judges, 3 requests, 201 services) and 7236 multi-dimensional judgments (4 judges, 3 requests, 201 services, 3 relevance dimensions) were derived this way. For the following presentation of results, please note that judges were asked to not use the ExcessMatch value on the Scope dimension of multi-dimensional relevance or for one-dimensional graded relevance. The services from JGD are exclusively data services. Since additional data can usually be filtered without harm, it was decided that these relevance values are not necessary for this data set.

7.6.2. Results

Results showed a large variance between the three different requests. This is not surprising also because the requests had explicitly been chosen to show varying characteristics from a very clear request with comparatively fewer matches (Location Distance) to a rather vague request with very few direct, but lots of partial matches (US City Data). Consequently, the different requests had largely different numbers of services judged relevant at the various relevance levels. This is illustrated by Figure 7.4 which shows the number of one-dimensional relevance judgments for

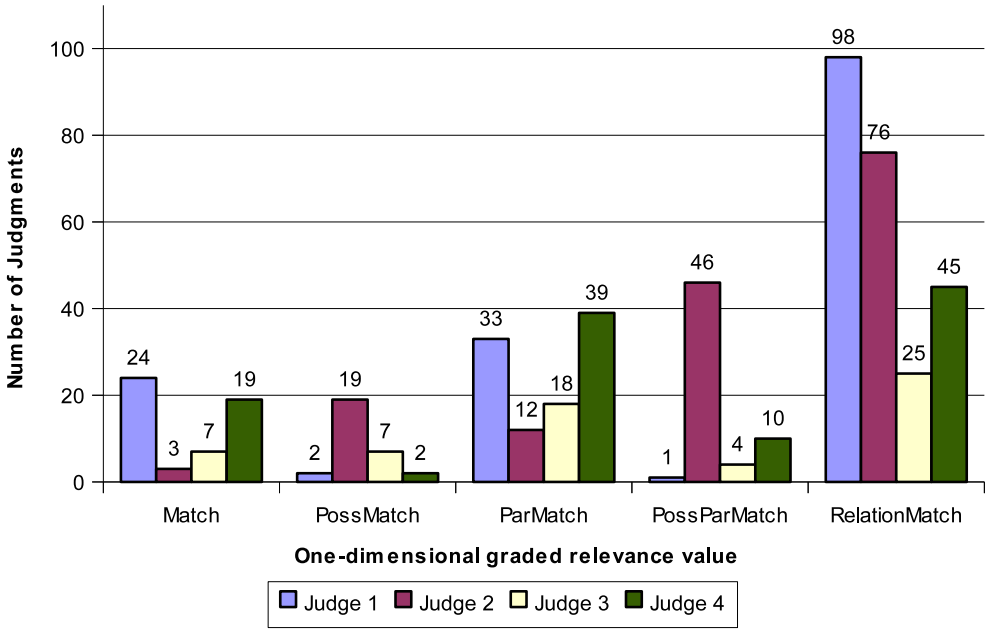

Figure 7.5.: Distribution of one-dimensional relevance values by judge

each relevance level and each request summed over the four judges (not including NoMatches, i.e. completely irrelevant services).

However, we also noted significant differences in the way how the four judges assessed the services. This is illustrated by Figure 7.5 which shows the number of judgments by each judge for each one-dimensional relevance level, summed over the three requests (not including NoMatches, i.e. completely irrelevant services). As can be seen, some judges were more liberal and judged more services relevant (e.g., Judge 1), others were much stricter and judged much fewer services relevant (e.g., Judge 3). Only Judge 2 made significant use of PossMatch and PossParMatch, thus making ambiguities in the service documentations explicit.

Inconsistency in Judgments

As is obvious from Figure 7.5, there was significant inconsistency in the judgments from different judges (this will be detailed below). However, there was even significant inconsistency in the judgments of any single judge, too. The multi-dimensional relevance judgments can be reduced unambiguously to one-dimensional (or binary) ones. The relevance judgments resulting from reducing the multi-dimensional ones can then be compared with the original one-dimensional judgments by the same

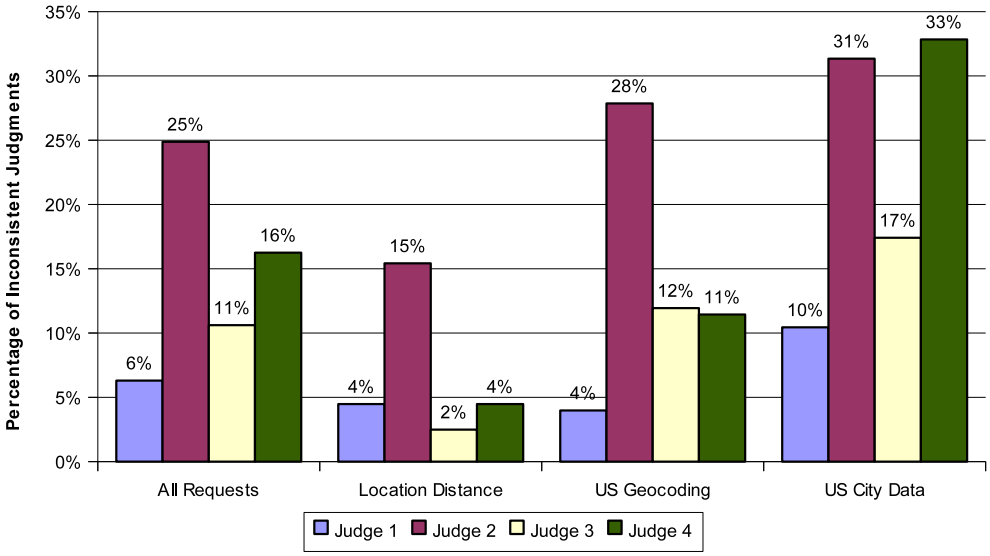

Figure 7.6.: Inconsistency in multi-dimensional and one-dimensional relevance judgments by the same judge

judge to determine their consistency. Figure 7.6 shows the inconsistency between original one-dimensional judgments and those obtained from reducing the multi-dimensional ones for the four judges and the different requests. It illustrates a sometimes surprisingly high inconsistency which, however, varies significantly from judge to judge (e.g., Judge 1 judged significantly more consistent than Judge 2). Again, there is also a large variation depending on the request at hand (e.g., Location Distance versus US City Data). This is not really surprising, given the different characteristics of the requests discussed above.

This variation was also displayed in the inter-judge inconsistency which for four judges, for instance, varied (depending on the used relevance scale) from 10%-20% for the Location Distance request, but 34%-53% for the US City Data request. Figure 7.7 presents a more comprehensive illustration of the disagreement of judgments by different judges. It shows the percentage of inconsistent judgments by two, three, and four different judges using binary, one-dimensional (Graded) as well as multi-dimensional (MD) relevance. The disagreement in the multi-dimensional judgments has been measured in two variants. The strict version (MD-Strict) considers judgments consistent only if they agree on all dimensions. The relaxed version (MD-Relaxed) counts the consistent judgments on the dimensions independently. I.e., if judges agreed on two of the three dimensions for a service-request pair, the relaxed version shows a disagreement of 33%, but the strict version one of 100%. Finally,

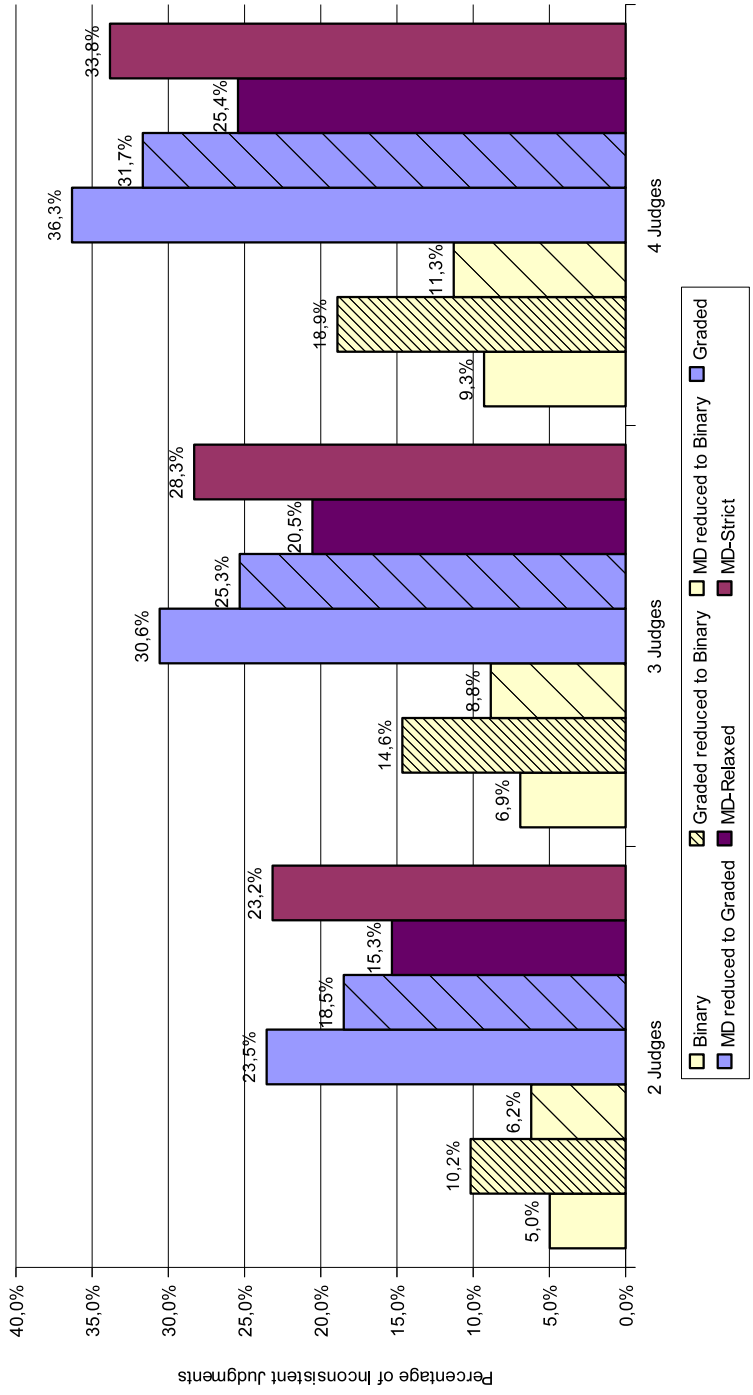

Figure 7.7.: Disagreement in relevance judgments by different judges

Figure 7.7 also displays the level of disagreement in the multidimensional and one-dimensional judgments measured after they have been reduced to one-dimensional and binary judgments. In all cases the values shown for the agreement among two and three judges are the averages of the values for all possible choices of two respectively three out of the four judges.

Please note that counting inconsistent judgments does not differentiate between different levels or qualities of disagreement. I.e., using four judges, it does not matter whether only one judge deviated only slightly from the other three judges or whether each of the four judges presented an entirely unique judgment. Such different qualities of disagreement could be quantified using measures like variance. However, measuring the variance requires the underlying scale to be an interval scale (where the difference between two measures is meaningful), whereas the relevance judgments were not even strictly ordinal (it is not necessarily clear, whether, for instance, a RelationMatch is preferable to a PossParMatch or not) [WRH+00, Chap. 3.1 and 8.1]. Thus, we resorted to simply counting the number of inconsistent judgments without further distinction.

Not surprisingly, the number of inconsistent judgments increases, if more judges are taken into account and need to agree. This is consistent with previous similar experiments in IR, albeit our results show higher consistency and a much less drastic deterioration of consistency when adding more judges than typically reported [Sar08].

However, the overall levels of agreement, even though higher than reported in most IR experiments, were still much lower than we had expected. Disagreement among judges can be attributed to either different interpretations of insufficiently precise service documentations or simply to objectively incorrect judgments. While it is not easy to make that distinction in all cases, it seems that by far most cases of inconsistent judgments can be attributed to objectively incorrect judgments. One judge, for instance, generally ignored a judgment instruction to disregard license information (whether a user name or license code was required as service input) when judging the relevance of a service. Furthermore, a very small difference in the service documentation, e.g., a small note at a service output of type address which states whether the address is provided with or without geographic location, often determines whether or how much a services is relevant. Given the large number of judgments that the relevance judges had to provide, errors resulting from a lack of concentration or diligence were apparently quite common.

Comparison of Relevance Scales

Comparing the inconsistency observed with different relevance scales, it seems that using multi-dimensional relevance reduces the number of errors compared to using one-dimensional relevance. Figure 7.7 shows that even if strictly measured, the

usage of multi-dimensional relevance results in slightly less disagreement than that of one-dimensional graded relevance. This is remarkable. The multi-dimensional relevance scheme allows for 75 different ways to judge a service, 40 of which were actually used. The one-dimensional scale allows for only six different ways to judge a service (keep in mind that ExcessMatch was not used). Therefore one would expect more disagreement in the multi-dimensional relevance scheme, but as mentioned, the opposite was the case. We believe that the usage of multi-dimensional relevance supported the judges in building their decision in a more careful and structured way, thus leading to fewer errors and slightly higher instead of lower consistency.

This interpretation is also supported by the fact that one-dimensional judgments resulting from reducing the multi-dimensional ones (blue bars with diagonal lines in Figure 7.7) showed notably less disagreement than the original one-dimensional ones (plain blue bars). Similarly, the binary judgments resulting from reducing the multi-dimensional ones (yellow bars with wide diagonal lines) also showed significantly less disagreement than the binary judgments resulting from reducing the one-dimensional ones (yellow bars with narrow diagonal lines). However, both, in particular the latter ones, showed more disagreement than the directly created binary judgments (plain yellow bars). We hypothesize that this is primarily due to the fact that the binary judgments were added later, thus created after the judges had already judged the test collection twice. It appears that the knowledge gained about the test collection during the previous rounds of judgments resulted in more consistent judgments.

7.6.3. Conflict Resolution and Consensus Building

As mentioned, the encountered inconsistency in judgments was higher than expected and not satisfying. In order to address this problem, we performed a conflict resolution phase. During this phase, judges were able to comment their judgments, to see the judgments and judgment comments of other judges, to search for services with conflicting judgments, to change their judgments and to set a status for their judgments. Admissible values for the status where as follows:

1. *Confirmed*
2. *Debatable - Guidelines ambiguous*
3. *Debatable - Service ambiguous*
4. *Revoked - Misinterpreted service*
5. *Revoked - Misinterpreted guidelines*
6. *Revoked - Judgment error.*

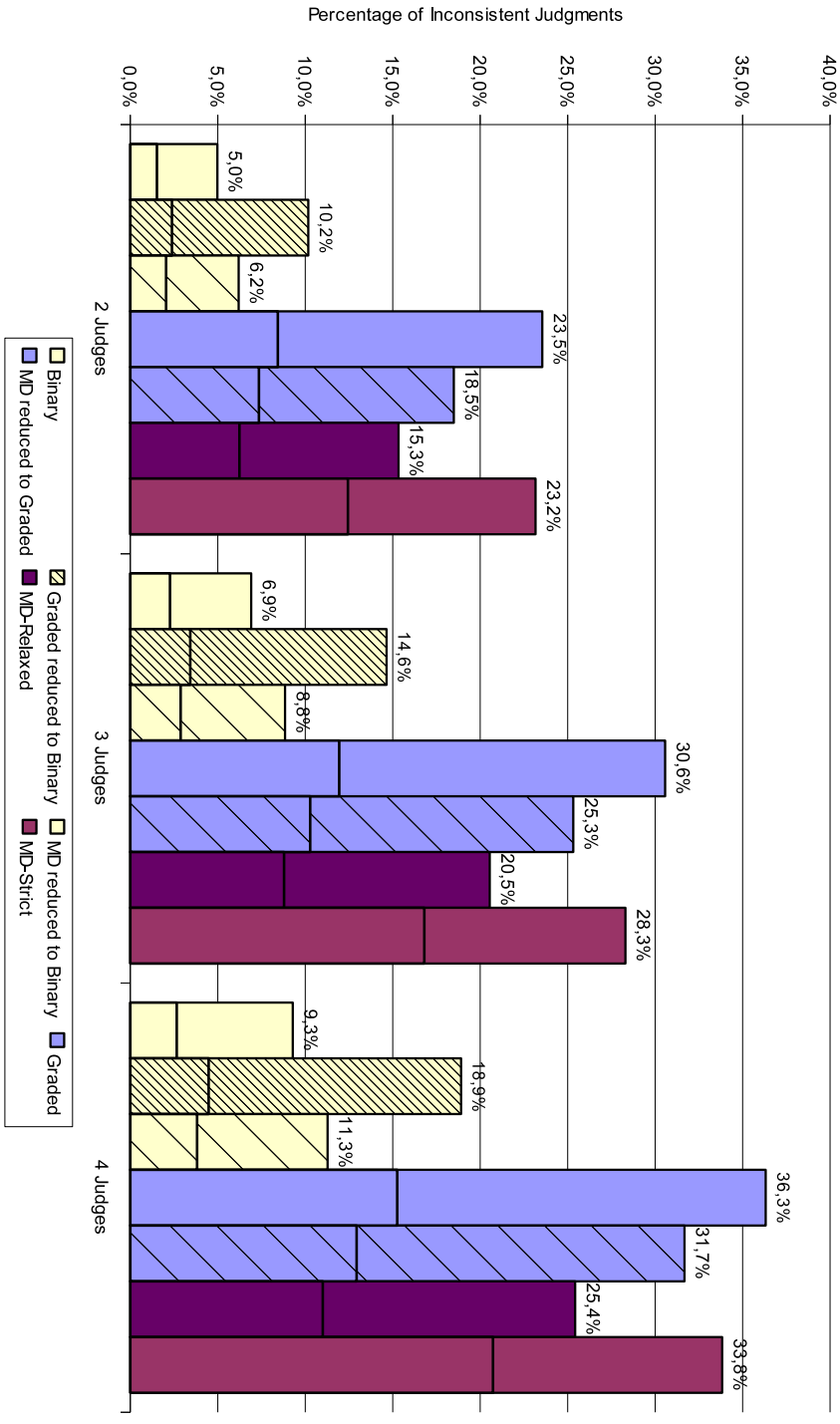

Figure 7.8.: Disagreement in relevance judgments by different judges before (full bars) and after conflict resolution (lower part of bars)

Status	J1	J2	J3	J4	Total
Confirmed	200	39	50	4	293
Debatable (Service ambiguous)	80	10	6	45	141
Debatable (Guidelines ambiguous)	4	23	3	0	30
Revoked (Misinterpreted service)	47	213	65	122	447
Revoked (Misinterpreted guidelines)	8	44	26	1	79
Revoked (Judgment error)	8	117	155	256	536

Table 7.1.: Usage of conflict resolution status values by judges

If a judge altered a judgment, one of the *Revoked* values had to be set, otherwise setting of a status was optional.

Table 7.1 shows the usage of status values during the conflict resolution phase. As can be seen, most judgment changes resulted from judgment errors (536) or misinterpretations of the services (447). A much lower number resulted from a misinterpretation of the judgment guidelines (79). This seems to imply that judges were rather confident about their understanding of the relevance scales. This is further supported by the number of judgments marked debatable. Here, 141 debatable judgments resulted from ambiguous services, but only 30 from ambiguities in the judgment guidelines.

All in all, almost ten percent of the judgments were changed. Naturally, this resulted in a significant reduction of inconsistency in judgments. Figure 7.8 shows the disagreement in relevance judgments by different judges before the conflict resolution phase (full bars corresponding to Figure 7.7) and after it (lower part of bars). As can be seen, the disagreement fell significantly to levels between 23% (Graded reduced to Binary, 4 Judges) and 61% (MD-Strict, 4 Judges) of the previous levels. The usage of status values discussed above suggests that this reduction corresponds to the correction of objectively wrong judgments primarily originating from a lack of concentration during the judgment process or other judgment errors. This also suggests that the majority of remaining inconsistent judgments resulted from the inherent subjectivity of a relevance judgment process that can not be resolved easily.

In order to create a set of final judgments, the judges met after the conflict resolution phase, discussed the remaining inconsistent judgments and agreed upon a set of consensus judgments. This resulted in further clarifications of the judgment guidelines, the reformulation of the US City Data Request, disambiguations of a few services and agreement on some domain assumptions, e.g., whether zip codes and city names provide an equally precise determination of a location or not.

We are very optimistic that the final consensus judgments represent a set of high quality judgments even though they may still contain a few flawed judgments. The process described above detects false or controversial judgments by introducing

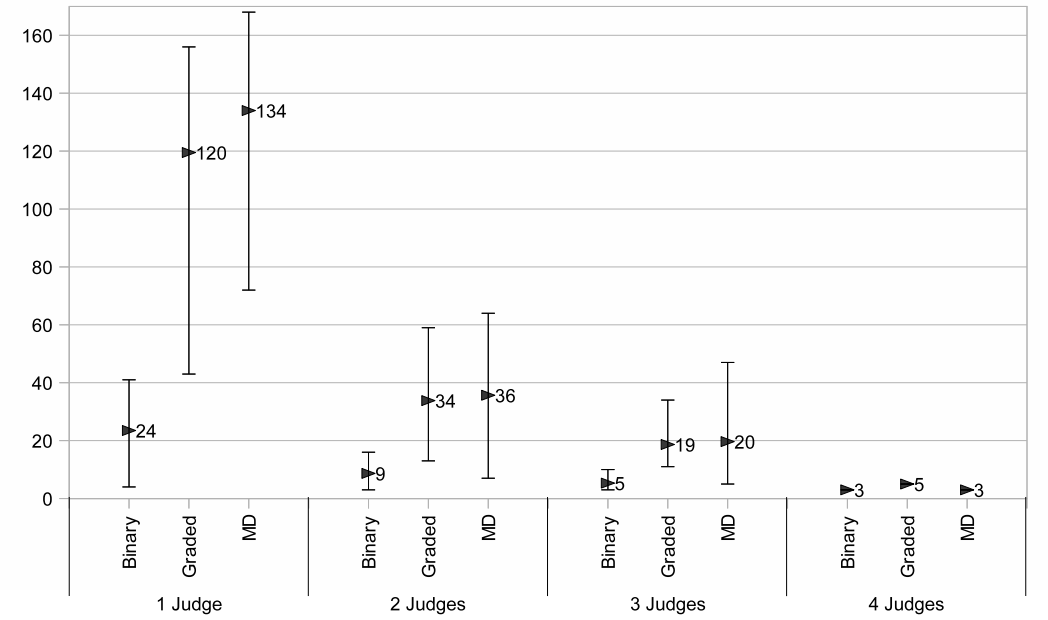

Figure 7.9.: Relationship between number of judges and detected judgment errors

redundancy in judgments and testing for inconsistencies in those redundant judgments. An interesting matter is the relationship between the number of redundant judges and the number of false or controversial judgments detected. Figure 7.9 shows the number of service-request pairs whose judgments would differ from the consensus judgments if less judges had been used. This is based on the assumption that, regardless of the number of judges used, original judgments remain unchanged through the process if they are consistent. The figure shows that if, for instance, only two judges had judged the collection using multi-dimensional relevance, between 7 and 64 service-request pairs (depending on the choice of two out of the four judges we used, 36 on average) would have had consistent judgments that are different from our final consensus judgments. Note that even using four judges, a few judgments were changed only because they were inconsistent with respect to different relevance scales, not different judges.

7.6.4. Conclusions

A number of conclusions may be drawn from the experiment even though some of them need to be taken with a grain of salt, since the number of requests that we ran the experiment with was small.

Significant inconsistency of relevance judgments is a problem long known in the IR community. Our experiment showed that the problem remains in Web service retrieval despite the much more restricted domain. We found that the problem of inconsistent judgments can be effectively addressed by using redundant judgments and a multi-phased approach for obtaining reference judgments. Interestingly, this is in contrast to conclusions from the IR community. The first IR retrieval performance study ever (Gull 1956) used redundant reference judgments and discovered that these were highly inconsistent. A reconciliation of the judgments failed and the study was not completed [Gul56]. Saracevic remarks with this respect: "The collapse of Gull's study influenced Cleverdon's selection of the method for obtaining relevance judgments, as it did every IR test done since then. The lesson was learned: Never, ever use more than a single judge (or a single object, such as source document) for establishing the gold standard for comparison. No test ever does" [Sar08].

There are two explanations for our conclusion differing from the cited one. First, SWS retrieval is a much more restricted domain, allowing to define characteristics of relevant service sufficiently precise to make a consensus building feasible. Second, establishing the reference relevance judgments is responsible for most of the work involved in building an IR test collection. In contrast, the main effort in building a SWS test collection is creating SWS descriptions. Consequently, in IR the problem of inconsistent judgments is primarily addressed by scaling up test collections such that inconsistent judgments affect the evaluation results only marginally (see Section 7.9.2). This approach is currently not feasible for SWS retrieval. Thus, spending additional effort on the reference judgments is much more reasonable in the area of SWS retrieval than in the area of general IR.

Besides this difference, our findings largely confirm previous findings from IR. Consistency of relevance judgments highly depends on the choice of test data. A data set with lots of partial matches naturally results in higher disagreement among judges than a data set with few precise matches. With this respect, the data set we used can be considered hard since the domain was rather specific (all services were geography and geocoding services) and the requests were selected such that there were many partial matches. Obviously, using a broader data set would result in a much larger number of clearly irrelevant services and thus overall higher consistency in judgments.

Naturally, inconsistency and noise increases if more relevance values are available. On the other hand, judgments on a scale with more values are more informative. Our definition of binary relevance, for instance, missed 65% of the service-request pairs judged somewhat relevant by the graded and multi-dimensional relevance scales. Comparing the consistency of judgments after the more powerful judgments have been reduced allows getting an impression of the trade-off between the increase in information and inconsistency (see Figure 7.8). With this respect, our experiment

clearly suggests that the multi-dimensional relevance scale is superior to the one-dimensional one.

A comparison of the multi-dimensional and the binary relevance scale is more difficult. The decrease of consistency seems roughly proportional to the gain in information. We did not feel that obtaining the multi-dimensional judgments involved significantly more effort per judge than obtaining the binary ones. However, Figure 7.9 suggests that two judges are probably sufficient to obtain reliable binary judgments, whereas three or four judges seem desirable when using multi-dimensional judgments. This seems to be the real price of the more informative multi-dimensional judgments.

While multi-dimensional graded relevance judgments clearly contain more information and are more flexibly useable during evaluations, it is yet to be shown that this actually results in more useful and reliable matchmaker evaluations. Furthermore, while we have shown that inconsistency is present in SWS retrieval reference judgments, it is still unclear, whether this also affects evaluation results. Both issues will be addressed in Section 7.9.

7.7. Retrieval Correctness Measures

The previous sections discussed issues around relevance and relevance judgments for SWS retrieval evaluation. This section completes this discussion by covering performance measures for SWS retrieval evaluation. Since the benchmark is primarily concerned with retrieval correctness evaluation, we will focus on corresponding measures in this section.

For an evaluation of runtime performance, measures are well understood and relatively straightforward. For the other evaluation criteria partially of interest within the scope of the benchmark, namely usability and coupling, simple, straightforward measures, like the time necessary to provide the required semantic annotations, or the amount of information being shared are also readily available. Desirable more advanced measures concerning for instance the perceived user satisfaction, the required knowledge and training of users or the complexity and maintainability of information being shared are beyond the scope of this work which, as mentioned, focuses on retrieval correctness evaluation.

Retrieval correctness measures need to provide a measure that allows quantifying the quality of the different rankings produced by the various matchmakers. This section will start with clarifying a few terms and discussing desirable characteristics for such measures. Afterwards, common measures from IR based on binary as well as graded relevance will be discussed. Finally, the measures will be discussed against the requirements and an improved set of measures will be proposed.

7.7.1. Basic Definitions and Desirable Measure Characteristics

The following definitions will be used throughout this chapter.

Definition 7.1 (Ranking) *A* ranking r *of a set of services* S *is an ordered sequence of the elements from* S*, i.e.:*

$$r = (r_1, r_2, \ldots, r_n), \quad n <= |S|, \quad r_i \in S, \quad r_i = r_j \Rightarrow i = j.$$

The number i *is called the* rank *of the service* r_i *with respect to the ranking* r*. A ranking with* $n = |S|$ *is called a* full ranking*.*

Definition 7.2 (Gain) *The* gain g *of a service* s *with respect to a query* q *denotes the relevance of* s *to* q*. We require the gain to be positive (*$g_q(s) >= 0, \forall s \in S$*). The function* g_q *which assigns each service* s *from a ranking* r *a gain* g *with respect to a query* q *is called a* gain function*. We furthermore define a binary flag that denotes whether a service at a given rank in a given ranking is relevant or not:*

$$isrel_{r,q}(i) = \begin{cases} 1 & : \quad g_q(r_i) > 0 \\ 0 & : \quad g_q(r_i) = 0 \end{cases}$$

Please note that $g_q(r_i) \in \{0, 1\}, \forall r_i \in S$ denotes the special case of binary relevance. For the sake of simplicity, we will generally omit the query index q and the ranking index r in the following if the query or ranking under consideration is clear from the context or no specific query or ranking is referenced.

Definition 7.3 (Ideal ranking) *A full ranking* r *is called* ideal *iff it lists the services in decreasing order of relevance, i.e.:* $\forall i \in \{2..|S|\} : g(r_i) <= g(r_{i-1})$*.*

Definition 7.4 (Retrieval effectiveness measure) *A* retrieval effectiveness measure M *is a function which assigns a ranking* r *a value from* $[0, 1]$ *with respect to a gain function* g*:* $M_g(r) \rightarrow [0, 1]$*.*

Having introduced a basic notion of retrieval effectiveness measure, we now turn to defining desirable properties of such measures. Again, a few definitions are helpful.

Definition 7.5 (Ranking superiority) *A ranking* r *is called* superior *to a different ranking* r' *(*$r > r'$*) iff* r' *can be changed into* r *by a sequence of pair wise item swaps within* r' *and for each two swapped items* r_i *and* r_j *the following holds:* $i < j \Rightarrow g(r_i) < g(r_j)$*. This corresponds to the intuitive notion that each swap improves the ranking.*

Definition 7.6 (Measure correctness) *A retrieval effectiveness measure* m *is called* correct *iff for any two different rankings* r *and* r'*,* $r > r' \Rightarrow m(r) > m(r')$ *holds.*

Ranking superiority and measure correctness formalize the intuitive notion that a ranking that lists items of higher relevance at comparatively higher ranks should always receive a superior performance score. Naturally, correctness in the given sense already makes assumptions about the use case underlying an evaluation. It has been argued, for instance, that the order of *relevant* items only may matter more than their absolute rank, i.e., the order of *all* items. This is based on the assumed model of a user that stops browsing an output ranking after having seen the first highly relevant item [SR08]. However, we argue that correctness in the given sense is most reasonable for the general retrieval use case defined at the outset of this chapter and thus require retrieval effectiveness measures to be correct. Besides this notion of correctness, three more properties of retrieval measures are desirable.

First, to avoid normalization issues that render an averaging of results over queries unstable, we require that an ideal ranking always receives a performance score of 1.

Second, for graded relevance, measures should allow to configure the extent to which an item of comparatively higher relevance is preferred over an item of comparatively lower relevance.

Third, for typical retrieval tasks, performance at the beginning of the output ranking is more important than performance at the end of the output ranking. For illustration, consider the following two rankings of items of binary relevance (1 denoting a relevant item and 0 an irrelevant one):

$$RankingA = (1, 0, 0, 0, 0, 0, 0, 0, 1)$$
$$RankingB = (0, 1, 0, 0, 0, 0, 0, 1, 0)$$

Please observe that neither A is superior to B nor vice versa. Still, A could be considered preferable over B since it performs better at the beginning of the ranking and a user may not even observe the difference in performance at the end of the output, especially if the outputs are even longer. Thus, from a user perspective, the quality of the ranking at its beginning is typically substantially more important than that at the end of the ranking. As will be discussed in more depth below, quantizing this *substantially* constitutes the main problem in choosing or designing a good retrieval evaluation metric. A good retrieval measure should thus emphasize top rank performance over bottom rank performance and allow to configure the extent of this emphasis.

7.7.2. Measures Based on Binary Relevance

After having briefly discussed desirable properties of retrieval effectiveness measures, we now turn to recalling some well established measures from IR. A complete coverage of such measures is beyond the scope of this chapter, but available in the standard IR literature, e.g. [BYRN99].

IR retrieval effectiveness measures are almost exclusively based upon the well-known *Recall* and *Precision* measures already mentioned at the outset of this chapter. Let R be the set of relevant items for a query. Let L be the set of items returned in response to that query. Then *Recall* is defined as the proportion of (binary) relevant items returned and *Precision* as the proportion of returned items that are (binary) relevant:

$$Recall = \frac{L \bigcap R}{R}, \ Precision = \frac{L \bigcap R}{L}.$$

Recall and Precision are set-based measures. However, there is an obvious trade-off between them. By returning more items, a system can usually increase its Recall at the expense of its Precision. Thus, in the following we assume that systems return a ranking, i.e. a list of items ordered by decreasing estimated confidence in relevance.

This allows measuring Precision as a function of Recall by scanning the output ranking from the top to the bottom and measure the Precision at standard Recall levels. These measures average well for different queries and the corresponding R/P charts are the most widely used measure to compare the retrieval performance of systems. It is also possible to measure Precision and Recall at a predefined rank l called *document cutoff level* ($Precision_l$ and $Recall_l$). However, these measures do not average well for queries where $|R|$ varies greatly since neither $Precision_l$ nor $Recall_l$ guarantee that an ideal ranking always receives a performance score of 1.

If a system's performance needs to be captured in a single measure, the most often used one is *Average Precision* over relevant items which is defined as:

$$AveP = \frac{1}{|R|} \sum_{i=1}^{|L|} isrel(i) \frac{count(i)}{i}.$$

Please observe that AveP is correct for binary relevance, always assigns a value of 1 to an ideal ranking and emphasizes top over bottom ranks ($AveP(RankingA) = 0.611$ but $AveP(RankingB) = 0.375$) even though it does not allow to configure the extent of the emphasis.

7.7.3. Measures Based on Graded Relevance

As mentioned previously, IR evaluation has primarily been based on binary relevance [BYRN99]. However, since about 2000, there is an increased interest in measures based on graded or continuous relevance [DM06, Kis05]. Various proposals have been made to generalize the Recall and Precision based measures from binary to graded relevance. For the sake of completeness, we briefly recall the most common ones.

All of them are based on or can be expressed in terms of *Cumulated Gain* proposed by Järvelin and Kekäläinen [JK02]. Intuitively, Cumulated Gain at rank i measures the gain that a user receives by scanning the top i items in a ranked output list. More formally, the *Cumulated Gain* at rank i is defined as:

$$CG(i) = \sum_{j=1}^{i} g(r_j).$$

Moreover, the *Ideal Cumulated Gain* at rank i, $ICG(i)$, refers to the cumulated gain at rank r of an ideal ranking.

Since $CG(i)$ can take arbitrarily large values for queries with many relevant items it has to be normalized to average or compare results across queries.

Normalized Cumulated Gain[6] at rank i is defined as the retrieval performance relative to the optimal retrieval behavior, i.e.:

$$NCG(i) = \frac{CG(i)}{ICG(i)}.$$

If we interpret NCG as the normalized cumulated gain at some document cutoff level l, we also write NCG_l. Normalized Cumulated Gain allows a straightforward extension of AveP which has sometimes been referred to as *Average Weighted Precision* [Sak04a]:

$$AWP = \frac{1}{|R|} \sum_{i=1}^{|L|} isrel(i) \frac{CG(i)}{ICG(i)}.$$

Unfortunately, NCG(i) has a significant flaw that AWP inherits: since ICG(i) has a fixed upper bound ($ICG(i) <= ICG(|R|)$), NCG(i) and AWP cannot penalize late retrieval of relevant documents properly since NCG(i) cannot distinguish at which rank relevant documents are retrieved for ranks greater or equal than |R| [Sak04b]. I.e., AWP is not correct in the sense defined above.

For illustration consider the following full rankings:

$$RankingC = (1, 0, 0, 1)$$
$$RankingD = (1, 1, 0, 0)$$

Clearly, B is superior to A (in fact, B is the optimal ranking), however, $AWP(A) = AWP(B) = 1$. Several measures have been proposed that resolve this flaw of AWP.

[6] A similar measure has already been proposed by Pollack in 1968 under the name *sliding ratio*.

Järvelin and Kekäläinen [JK02] suggested to use a discount factor to penalize late retrieval and thus reward systems that retrieve highly relevant items early. They defined *Discounted Cumulated Gain* at rank i as:

$$DCG(i) = \sum_{j=1}^{i} \frac{g(i)}{disc(i)}$$

with $disc(i) >= 1$ being an appropriate discount function. Järvelin and Kekäläinen suggested to use the log function and use its base b to customize the discount which leads to

$$DCG_{\log_b}(i) = \sum_{j=1}^{i} \frac{g(i)}{max(1, \log_b i)}.$$

We use an according definition of *Ideal Discounted Cumulated Gain* ($IDCG(r)$) to define the *Normalized Discounted Cumulated Gain* at some document cutoff level l ($NDCG_l$) and a corresponding Version of AWP that we call *Average Weighted Discounted Precision*:

$$AWDP = \frac{1}{|R|} \sum_{i=1}^{|L|} isrel(i) \frac{DCG(i)}{IDCG(i)}.$$

Kishida [Kis05] proposed a generalization of AveP that also avoids the flaw of AWP:

$$GenAveP = \frac{\sum_{i=1}^{|L|} isrel(i) \frac{CG(i)}{i}}{\sum_{i=1}^{|R|} \frac{ICG(i)}{i}}.$$

Not that GenAveP is similar to DCG, however, DCG applies a discount to the gain of an item before the gain is accumulated. In contrast, GenAveP applies a fixed discount ($disc(i) = i$) after cumulating gains.

Sakai [Sak04a] proposed an integration of AWP and AveP called Q-measure which inherits properties of both measures and possesses a parameter β to control whether Q-measure behaves more like AWP or more like AveP:

$$Q\text{-}measure = \frac{1}{|R|} \sum_{i=1}^{|L|} isrel(i) \frac{\beta CG(i) + count(i)}{\beta ICG(i) + i}.$$

Finally, it has also been proposed to use Kendall's τ or other rank correlation measures to measure retrieval effectiveness by comparing a ranking r with an ideal ranking r' [KG90, Mel07]. Kendall's τ measures the correlation between two rankings via the number of pair wise adjacent item swaps that are necessary to turn one

ranking into another. Since Kendall's τ yields values between 1 (identical rankings) and -1 (inverse rankings), it needs to be normalized to yield values from $[0, 1]$:

$$\tau'(r) = \frac{\tau(r, r') + 1}{2}.$$

7.7.4. Discussion of Measures

With the exception of τ', all measures introduced above allow fine-tuning the extent to which highly relevant items are preferred over less relevant items by choosing an appropriate gain function. Furthermore, except for CG_l and DCG_l all measures are properly normalized and assign an ideal ranking a score of 1. We now discuss the measures with respect to the other requirements, i.e., correctness and the degree of control that is possible with respect to the extent to which late retrieval is penalized. For illustration, consider the following full rankings of graded gain values:

$$\begin{aligned}
Ranking\ R_1 &= (10, 6, 3, 0, 0, 0, 0, 0, 0) \\
Ranking\ R_2 &= (10, 3, 6, 0, 0, 0, 0, 0, 0) \\
Ranking\ R_3 &= (6, 10, 3, 0, 0, 0, 0, 0, 0) \\
Ranking\ R_4 &= (3, 6, 10, 0, 0, 0, 0, 0, 0) \\
Ranking\ R_5 &= (0, 0, 0, 3, 6, 10, 0, 0, 0) \\
Ranking\ R_6 &= (0, 0, 0, 0, 0, 10, 6, 3, 0) \\
Ranking\ R_7 &= (0, 0, 0, 0, 0, 0, 10, 6, 3)
\end{aligned}$$

Please observe that R_1 is the optimal ranking and that $R_1 > \{R_2, R_3\} > R_4 > R_5 > R_6 > R_7$. Furthermore, R_2 should be considered preferable to R_3 since the single item swap compared to the optimal ranking occurs between ranks two and three for R_2 but between the higher ranks one and two for R_3. Table 7.2 shows the computed performance scores according to the various measures for the seven rankings.

AveP

Trivially, binary AveP can not distinguish among items of different relevance grades and is thus not correct, if graded relevance is used, e.g., $AveP(R_1) = AveP(R_2)$.

NCG

NCG_l only regards the number of relevant items and their degree of relevance, but not their ordering and is thus also not correct, e.g., $NCG_3(R_1) = NCG_3(R_2)$.

Measure	R_1	R_2	R_3	R_4	R_5	R_6	R_7
AveP	1.00	1.00	1.00	1.00	0.38	0.28	0.24
NCG_3	1.00	1.00	1.00	1.00	0.00	0.00	0.00
$NDCG_9$ ($disc(i) = \sqrt{i}$)	1.00	0.98	0.93	0.81	0.52	0.46	0.43
AWP	1.00	0.94	0.87	0.62	0.54	0.79	0.79
Q-measure ($\beta = 1$)	1.00	0.94	0.88	0.66	0.50	0.65	0.63
GenAveP	1.00	0.94	0.84	0.57	0.23	0.26	0.23
AWDP ($disc(i) = \sqrt{i}$)	1.00	0.94	0.81	0.54	0.29	0.37	0.35
τ'	1.00	0.97	0.97	0.92	0.67	0.58	0.50

Table 7.2.: Comparison of evaluation measures

Naturally, it also can not differentiate rankings that are identical up to rank i but differ at ranks greater than i.

NDCG

$NDCG_l$ resolves the first problem, if the discounting function $disc(i)$ is valid, i.e., positive and strictly monotonic increasing for $i \in [1, l]$. Notably, this is not the case for the originally suggested and typically used $max(1, \log_b(i))$ discount function which is constant for ranks 1 through b. With valid discounting functions, however, $NDCG_l$ is correct as far as rankings are only considered up to rank l. $NDCG_l$ also allows configuring the extent to which late retrieval is penalized by choosing a more or less quickly growing discount function.

AWP

AWP can not differentiate among rankings that are equal till rank |R|, e.g., $AWP(R_6) = AWP(R_7)$. This is the previously mentioned inability to properly penalize late retrieval beyond rank |R| that has been well discussed in the literature [Sak04a]. Even worse, for relevant items retrieved below rank |R|, the order of items matters more than their absolute rank, e.g., $AWP(R_5) < AWP(R_6)$, despite of $R_5 > R_6$. AWP is thus not correct. To the best of our knowledge, this defect has not been discussed so far. AWP also does not allow configuring the extent to which late retrieval is penalized.

Q-Measure

Q-Measure was designed to resolve the first defect of AWP, but unfortunately inherits the second one, e.g., $Q\text{-}measure(R_5) < Q\text{-}measure(R_6)$. The actual vulnerability of Q-Measure to this defect depends upon the actual choices for the gain values

and its β factor. But for any setting, it either inherits the vulnerability from AveP of not properly distinguishing among items of varying relevance or the vulnerability from AWP of rewarding superior ordering rather than superior ranking of relevant items. It is thus also not correct. Q-Measure provides some control over the extent to which late retrieval is penalized via its β factor.

GenAveP

GenAveP shares the order versus rank defect with Q-Measure and AWP, e.g., $GenAveP(R_5) < GenAveP(R_6)$. Therefore, just like Q-Measure and AWP, it is not correct. However, in practice, GenAveP seems to be somewhat less vulnerable to the mentioned defects than the other two measures. GenAveP does not allow configuring the extent to which late retrieval is penalized.

AWDP

AWDP resolves the first defect of AWP if the used discounting function is valid. Nevertheless it inherits the second AWP-defect of rewarding order rather than rank, e.g., $AWDP_{\sqrt{i}}(R_5) < AWDP_{\sqrt{i}}(R_6)$. It is therefore also not correct. Like the choice of β for Q-Measure, the choice of a discount function for AWDP has an influence on its practical vulnerability to this particular defect. By choosing a proper discount function AWDP allows configuring the extent to which late retrieval is penalized.

Rank Correlation Measures

Kendall's τ, respectively τ', is correct in the sense provided above. However, it does not differentiate between swaps that occur at the top and those that occur at the bottom of a ranking, e.g., $\tau(R_2) = \tau(R_3)$. Furthermore, as mentioned above, it also does not allow to configure the extent to which highly relevant items are preferred over less relevant ones.

Summary

It is remarkable that, as can be seen from this discussion, almost all commonly used evaluation measures based on graded relevance are not correct in the sense defined above. Of the discussed measures, the only ones that are correct are NDCG with a proper discount function and Kendall's τ. Still, NDCG is typically used with an improper discount function $(max(1, \log_b(i))$, effectively rendering it incorrect, too, and Kendall's τ has the important shortcoming of not being able to emphasize top versus bottom rank performance and also not being able to configure the extent to which highly relevant items are preferred over marginally relevant ones.

7.7.5. Proposed Improvements

After having discussed shortcomings of most commonly used measures for graded relevance, we now propose improvements for some measures to avoid these shortcomings. Table 7.3 shows a comparison of the original with the altered versions of the measures that illustrates how the altered versions avoid the problems of the original ones.

NDCG

The issues with NDCG can be trivially avoided by using an adapted version of the original discount function, namely $disc(i) = \log_b(i + b - 1)$, or any other valid function, like the root function, i.e., $disc(i) = i^a, 0 < a <= 1$. Such obvious adaptations have been proposed previously, e.g., [BSR+05]. Therefore, it is somewhat surprising to see that most literature still uses the original flawed discounting functions, e.g., [Sak07b].

AWP/AWDP

The defects of AWP and AWDP can be avoided by not averaging over relevant items only, but over all items, i.e.:

$$AWP' = \frac{1}{|R|} \sum_{i=1}^{|L|} \frac{CG(i)}{ICG(i)}$$

$$AWDP' = \frac{1}{|R|} \sum_{i=1}^{|L|} \frac{DCG(i)}{IDCG(i)}.$$

These measures can be interpreted as the area under a NCG- or NDCG-chart [JK02]. To properly distinguish them from the original versions, we will refer to them as *Averaged Normalized Cumulated Gain (ANCG)* and *Averaged Normalized Discounted Cumulated Gain (ANDCG)* in the following.

GenAveP

GenAveP can be fixed in the same manner as AWP/AWDP, i.e.:

$$GenAveP' = \frac{\sum_{i=1}^{|L|} \frac{CG(i)}{i}}{\sum_{i=1}^{|R|} \frac{ICG(i)}{i}}.$$

Measure	R_1	R_2	R_3	R_4	R_5	R_6	R_7
AWP	1.00	0.94	0.87	0.62	0.54	0.79	0.79
ANCG	1.00	0.98	0.96	0.87	0.51	0.37	0.26
GenAveP	1.00	0.94	0.84	0.57	0.23	0.26	0.23
GenAveP′	1.00	0.97	0.91	0.76	0.30	0.20	0.13
AWDP ($disc(i) = \sqrt{i}$)	1.00	0.94	0.81	0.54	0.29	0.37	0.35
ANDCG ($disc(i) = \sqrt{i}$)	1.00	0.96	0.89	0.72	0.27	0.18	0.12

Table 7.3.: Comparison of altered evaluation measures

Others

In contrast to the previous measures, Q-Measure can not be fixed in the same fashion. Recall that Q-Measure is an integration of AWP and AveP. Averaging over all, and not only relevant items, decreases the performance value of the AveP part of Q-Measure to values much smaller than 1.0 even for optimal rankings if the number of relevant items is much smaller than the total number of items. I.e., an optimal ranking for a query with many relevant items will still yield a performance close to 1.0, but an optimal ranking for a query with few relevant items will yield a performance much lower than 1.0, since AveP inevitably decreases once a ranking runs out of relevant items. This normalization effect makes averaging of results over queries with differing number of relevant items unstable.

Similarly the issues with Kendall's τ can also not been fixed easily. Rank correlation measures are simply not designed to distinguish between whether rankings differ at the top or bottom. Furthermore, rank correlation does not offer an intuitive way of configuring the extent to which highly relevant items are preferred over less relevant items.

7.7.6. Conclusions

The discussion above has shown that various measures for graded relevance are available, but that even some of the common ones behave unintuitive in certain cases. A fix for the problems associated with AWP, AWDP and GenAveP has been proposed. With this fix, NDCG, ANCG (fixed AWP), ANDCG (fixed AWDP) and GenAveP' (fixed GenAveP) are correct as defined above.

While this correctness guarantees a ranking of matchmakers which corresponds to intuition if the matchmaker's output rankings are pair wise superior, it does not guarantee a good ranking of matchmakers that produce outputs that are not pair wise superior, the common case in realistic settings. For such rankings, there is no obvious objective notion of superiority, since a decision has to be made how to balance highly relevant against less relevant items and performance in top ranks

against performance in lower ranks (or recall versus precision for that matter). Section 7.9.3 will thus complement the theoretic discussion in this section by an investigation of the behavior of the covered measures based on real rankings in a realistic retrieval experiment. Based upon this investigation, recommendations for retrieval effectiveness measures will be provided in Section 7.9.4.

7.8. Reference Execution of the Benchmark

After having discussed the theoretic background of the benchmarking setup presented in Section 7.4, we now report on a reference implementation of this approach. A corresponding evaluation campaign has been organized as a third track (JGD Evaluation) of the 2009 S3 Contest on Semantic Service Selection. Full information is available on the 2009 S3 Contest web site[7] as well as the web site of the JGD Evaluation[8]. This section describes the benchmarking event, including the dataset, the participating systems, the created service descriptions, the evaluation environment and the evaluation results. The following section will then analyze the evaluation results and discuss lessons that can be learned from it for the future.

7.8.1. Dataset

The dataset being used for the evaluation is the Jena Geography Dataset (JGD) which has been introduced in Section 5.4. In order to reduce the entry barrier to participation in the evaluation, the dataset has been divided into inclusive smaller datasets of 200 (full dataset), 150 (JGD150), 100 (JGD100), respectively 50 services (JGD50). It turned out that except for one, all participants chose to participate on JGD50, i.e., the smallest dataset.

To further reduce the necessary effort of participants, we created an OWL geography domain ontology based upon the PROTON ontologies[9]. Usage of this ontology was entirely optional.

Finally, the three queries discussed in Section 7.6 were complemented by seven more queries. For these, consensus relevance judgments according to the procedure described in Section 7.6 were obtained from three relevance judges according to the multidimensional relevance introduced in Section 7.5. Since one of the requests did not have matching services within JGD50, it was later removed, leaving nine service queries as the base of the evaluation. Further information about the queries will be provided in Section 7.8.4.

[7] http://www-ags.dfki.uni-sb.de/~klusch/s3/html/2009.html
[8] http://fusion.cs.uni-jena.de/professur/jgdeval/
[9] http://fusion.cs.uni-jena.de/professur/jgd

7.8.2. Participating Systems

The benchmarking event was publicized in the community on several mailing lists, through personal communication with groups active in the area and at relevant conferences. In the end, five groups participated with six service retrieval engines (the following brief descriptions have been provided by the developers of those engines).

Themis-S

by University Münster, Germany, is an IR-style service discovery engine, which can be regarded as a meet-in-the-middle approach between heavyweight semantic web technologies and easy-to-use syntactic information retrieval models [KKP08]. Themis-S is built upon the enhanced Topic-based Vector Space Model (eTVSM), an information retrieval model which is able to consider semantic relations in natural language text by exploiting the lexical semantics of a to be provided ontology. Themis-S parses natural language documents of service descriptions and requests and tries to extract concepts which are inherent in a provided ontology. From the bag of extracted concepts a document model is constructed which acts as a representation of the original natural language document for all subsequent steps. Analogous to the classical Vector Space Model (VSM) these document models can be regarded as vectors in a high-dimensional vector space. However, eTVSM does not suffer from the false assumption that two different terms are independent (orthogonal) of each other. In contrast to VSM, eTVSM operates on concepts (word meanings) rather than terms (words). Semantic relations (i.e. synonymy, homonymy, and hyponymy/hypernymy) between concepts can be modeled in a domain ontology. When constructing document models and calculating their similarity eTVSM exploits these semantic relations. The basic assumption is that the shorter the distance (path) between two concepts (nodes) in the ontology (graph) is, the higher their semantic similarity is.

WSColab

by University Modena and Reggio-Emilia, Italy, is a service retrieval engine based upon the folksonomy approach of collaborative tagging[10]. It distinguishes among tags for inputs, outputs and the overall behavior of a service. The matchmaking process returns services that are either interface compatible (service IO tags match at least one input query keyword and one output query keywords) or behavior compatible (there is a match with a behavior tag) to provide a possibility of matching related services with different interfaces. Matching services are ranked by combining a ranking in each dimension (input, output, behavior) using adapted standard

[10] `http://www.ibspan.waw.pl/~gawinec/wss/wscolab.html`

TF-IDF measure. For participation in JGDEval, an altered version of WSColab was used which returns the ranked matching services followed by all non-matching services in random order.

SAWSDL-MX1

by DFKI Saarbrücken, Germany, combines logic-based and syntactic (text similarity-based) matching to perform hybrid semantic matching of I/O parameters defined for potentially multiple operations of a Web service interface (signature matching) [KK08].

SAWSDL-MX2

by DFKI Saarbrücken, Germany, is an improved hybrid and adaptive version of the SAWSDL-MX1 matchmaker. It's logic component is based upon I/O concept subsumption relations (Equivalence, Plug-In, Subsumes, Subsumed-By). Ranking is performed in decreasing order of logical relation class. The non-logic component combines two similarity measures. First, the text similarity of unfolded I/O concept definitions processed into TFIDF keyword vectors over index and classical token-based text similairty measure cosine. Second, the structural similarity of WSDL groundings via recursive XML / WordNet based similarities of WSDL description elements. Both similarities result in numeric rankings in $[0, 1]$. MX2 is adapted offline via SVM-based binary relevance classifier with ranking via distance in $[0, 1]$ of considered service pairs to learned hyperplane in non-linearly separable 7-dimensional matching feature space [KKZ09].

SAWSDL iMatcher

by University Zurich, Switzerland, is a hybrid matchmaker based on input/output concepts and service name. SAWSDL-iMatcher 1 exploits a learned linear regression function to predict the similarity between a query and a service. The input variables of the linear regression model are the syntactic similarity of service name, the semantic similarity of input concepts and the semantic similarity of output concepts. SAWSDL-iMatcher 2 does not use the learned model but just average these three similarity values. For participation in JGDEval, SAWSDL-iMatcher 1 trained on the SAWSDL-TC used in Track 2 of the S3 Contest was used.

IRS-III

by Open University, UK, is an ontology-based reasoning and SWS broker environment based on OCML and LISP[11] [CDG+06]. It uses a SWS model compliant with

[11] `http://technologies.kmi.open.ac.uk/irs/`

the Web Service Modeling Ontology WSMO. OWL/RDF and WSML are translated to IRS-III's internal OCML/LISP format. For participation in the JGDEval, a preliminary implementation of a new discovery algorithm was used instead of IRS-III's typical goal invocation mechanism (which solves a goal by finding and invoking a suitable web service instead of delivering a ranked list of probably matching services). Domain ontologies are used to define service input and output types whose inheritance structure is used for the matchmaking. Currently, matching results are binary. The version used in the JGDEval returns the matching services followed by the non-matching services in random order.

7.8.3. Service Descriptions

To relieve participants from most of the tedious copy and paste work involved in creating semantic annotations for the JGD, we semi-automatically generated description templates from the structured information available for JGD according to template definitions that the participants provided. This allowed them to efficiently and smoothly integrate the JGD information into their development environments.

Participants then created service descriptions for the JGD according to the process defined in Section 7.4. They were asked to report the amount of effort it took them to create the service descriptions, but only little feedback on this was given. The created descriptions are available online[12] and will be described here only briefly.

Themis-S: For Themis-S, English natural language service descriptions were fully automatically generated from the data available in the JGD via the OPOSSum portal. Themis-S could have been used on the full dataset, but for the sake of comparison with other systems, we also restricted the dataset to 50 services.

WSColab: For WSColab, XML representations of all available information about the services were generated. This information was then presented to users in an online portal who chose free text tags they deemed suitable for the services. This way, 1540 tags were collected. Additionally, 1291 tags present in OPOSSum for the services were added. These tags were then represented in XML files describing the services.

SAWSDL-MX1 / SAWSDL-MX2: The SAWSDL descriptions were based upon the WSDL descriptions available for JGD. IO types from these WSDLs were manually linked to concepts from OWL domain ontologies (including but not limited to the geography domain ontology provided with JGD).

SAWSDL-iMatcher: SAWSDL-iMatcher reused the descriptions created for SAWSDL-MX1 and SAWSDL-MX2 without changes.

[12] `http://fusion.cs.uni-jena.de/professur/jgdeval/jgdeval-at-s3-contest-2009-results`

MD Judgment	Binary 1	Binary 2	Binary 3	Binary 4
Equivalence at least	PossEqual	PossEqual	Approximate	Approximate
Scope at least	PossMatch	Partial	PossMatch	Partial
Interface at least	PossCompatible	PossCompatible	PossCompatible	PossCompatible

MD Judgment	Binary 5	Binary 6	Binary 7	Binary 8
Equivalence at least	PossEqual	Approximate	Approximate	PossEqual
Scope at least	Partial	PossMatch	Partial	PossMatch
Interface at least	Incompatible	Incompatible	Incompatible	Incompatible

Table 7.4.: Binary relevant services for the JGD Evaluation

IRS-III: For IRS-III, the structured service information were automatically serialized to appropriate LISP/OCML templates according to a structure specified by Open University. The templates contained in particular semi-unique references for input and output types. These IO types were then manually mapped to concepts from a domain ontology. The employed domain ontology is a version of the provided geography ontology translated to OCML.

7.8.4. Evaluation Environment

The evaluation was performed with the Semantic Matchmaking Evaluation Environment SME2 2.1 revision 1 which has been provided by DFKI Saarbrücken, Germany, and is available online at SemWebCentral[13]. SME2 defines an interface which matchmakers have to implement to make them pluggable into the evaluation environment. Participants provided this interface and the binaries of their matchmakers. Service descriptions and relevance judgments were assembled to test collections according to the format defined by SME2. SME2 then automatically sends the proper service descriptions to the matchmakers, queries them with the predefined request descriptions, measures the execution time and stores the returned rankings as well as a number of performance measures. However, in order to be able to efficiently switch different evaluation settings and also analyze performance measures currently not supported by SME2, the analysis of the stored output rankings was performed outside of SME2.

The retrieval effectiveness measures described in Section 7.7 allow to evaluate SWS retrieval systems based on binary or graded relevance, but leave open the question about the proper parameter combinations to use in an evaluation. As Järvelin and Kekäläinen remark, "the mathematics work for whatever parameter combinations and cannot advise us on which to choose. Such advise must come from the evaluation context in the form of realistic evaluation scenarios" [JK02]. In

[13] `http://projects.semwebcentral.org/frs/?group_id=150`

MD Judgment	Graded 1	Graded 2	Graded 3	Graded 4
Equivalence	0, 1, 2, 3, 4	0, 2, 4, 6, 8	0, 1, 2, 3, 4	0, 1, 2, 3, 4
Scope	0, 1, 2, 3, 4	0, 1, 2, 3, 4	0, 2, 4, 6, 8	0, 1, 2, 3, 4
Interface	0, 2, 4	0, 2, 4	0, 2, 4	0, 4, 8

Table 7.5.: Graded relevance settings for the JGD Evaluation (gain values)

order to perform an investigation of the effects of switching from binary to graded relevance and the sensitivity of measures to changes in the relevance definition, we performed the evaluation using eight binary and four graded relevance settings.

The binary relevance settings were created by reducing the multi-dimensional graded reference judgments to binary ones according to the settings specified in Table 7.4. With the Binary 2 setting, for instance, services judged at least PossEqual on the Equivalence dimension, at least Partial on the Scope dimension and at least PossCompatible on the Interface dimension were considered binary relevant, all others binary irrelevant.

For measures based on graded relevance we assigned a gain value to each relevance value on each dimension. A service was then assigned the combined gain values of the three dimensions. We used four different settings: A balanced one and each one emphasizing Equivalence, Scope and Interface respectively. Table 7.5 shows the gain values associated to the single dimensions according to the four different settings. E.g., a service judged Approximate (Equivalence), PossMatch (Scope) and Compatible (Interface) receives a gain of 9 ($2 + 3 + 4$, Graded 1), 11 ($4 + 3 + 4$, Graded 2), 12 ($2 + 6 + 4$, Graded 3) and 13 ($2 + 3 + 8$, Graded 4).

Finally, Table 7.6 provides an overview of the number of binary matching services for each of the eight binary relevance settings and each of the nine queries. The last column contains the number of relevant services (gain value positive) for the four graded settings. By definition, the number of services with a positive gain is equal for all four graded relevance settings. Please note that for strict binary settings, there were queries with no matching services, caused by the fallback from the full JGD to JGD50. Having requests without matching services is not generally harmful (all matchmakers benefit equally from such requests) but effectively reduces the basis of the evaluation since effective data can only be retrieved from requests with matching services.

7.8.5. Evaluation Results

In the following, we will briefly summarize the evaluation results. However, in the context of this chapter, the results, especially those about retrieval correctness, are primarily of interest with respect to the question of what can be learned from them

Request	Bin 1	Bin 2	Bin 3	Bin 4	Bin 5	Bin 6	Bin 7	Bin 8	Graded
Query 1	0	2	0	2	2	0	2	0	2
Query 2	0	2	0	2	8	4	11	4	11
Query 3	2	3	2	3	3	2	3	2	3
Query 4	2	16	2	16	32	2	38	2	38
Query 5	13	15	13	15	18	23	25	16	25
Query 6	12	13	23	25	13	23	25	12	25
Query 7	2	3	3	4	7	7	8	6	8
Query 8	12	13	12	13	17	16	17	16	17
Query 9	17	17	17	17	22	23	25	21	25

Table 7.6.: Relevant services by query and relevance setting

about this benchmarking approach. A corresponding discussion will be provided in Section 7.9.

Description Effort and Coupling

As mentioned, we attempted to gather data on the effort for creating the necessary service descriptions and domain ontologies but the feedback from the participants was not very detailed. Apparently participants did not record their effort or were reluctant to disclose it. Nevertheless, the following observations can be made:

- The descriptions processed by Themis-S were generated completely automatically.
- The taggings used by WSColab were collected in an open environment and it is unclear, how much time the users spend tagging services. However, the average time to tag a service is probably rather low.
- The SAWSDL descriptions and IRS-III descriptions were created by manually linking IO types to concepts from a provided domain ontology. The exact effort it took to perform this task is unclear, but the effort is probably moderate.
- None of the participants used full-fledged logic descriptions with complex axioms for any pre- or post conditions of the services.

Regarding the relationship between retrieval efficiency and description effort, for this test collection and the evaluated matchmaker implementations WSColab and Themis-S probably present the best tradeoff (see retrieval correctness further below). However, this statement can not be generalized to the retrieval principles underlying the various matchmakers until confirmed by more experimentation.

Regarding coupling, the SAWSDL and IRS-III descriptions shared the common ontologies among service and request annotators. Similarly, users formulating search

Matchmaker	Total exec. time	Avg. query time
IRS-III	25.5 s	2826 ms
SAWSDL-MX1	4.41 s	162 ms
SAWSDL-MX2	11.05 s	785 ms
SAWSDL iMatcher	1.66 s	177 ms
Themis-S	99.97 s	2043 ms
WSColab	0.125 s	0 ms

Table 7.7.: Runtime performance results of the JGD Evaluation

tags to query WSColab had access to the tag clouds formed during the process of tagging the services. Themis-S did not share any information among requesters and providers except for common background knowledge (WordNet) which was not created explicitly for this experiment.

Runtime Performance

Runtime measurements were provided by Patrick Kapahnke, DFKI Saarbrücken, Germany and measured on an Intel Core2 Duo T9600 (2.8GHz) machine with 4 GB RAM running Windows XP 32bit. They are displayed in Table 7.7. No measures of the memory consumption are given, since the memory consumed by the matchmaker implementations outside of the SME2 evaluation environment was not traced.

Retrieval Correctness

Figures 7.10 and 7.11 illustrate the retrieval correctness results from the evaluation. Figure 7.10 shows the macro averaged binary precision at standard recall levels for the relevance setting Binary 7, i.e., the most relaxed definition of binary relevance. Figure 7.11 shows the normalized discounted cumulated gain using discounting function $\log_2(i+1)$ and relevance setting Graded 1. As mentioned, the concrete results are of minor importance for this chapter and primarily provided to give a context for the following discussion.

The following remarks about the results need to be made. First, WSColab used five different query formulations resulting in five sets of rankings. All following results for WSColab are obtained by averaging the results obtained for those five rankings.

Second, SAWSDL-MX2 is an adaptive matchmaker which was trained on 20% of the test collection (20% of all service-request pairs were randomly selected). Because of the limited size of the test collection, the training data had to be included in the final evaluation run. Thus, SAWSDL-MX2 had an advantage over the other

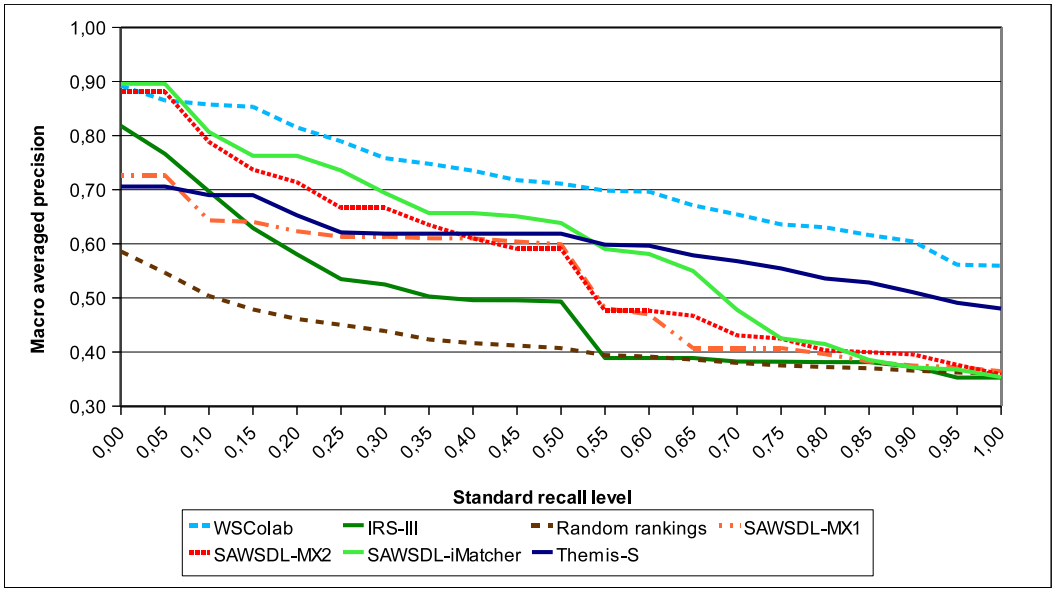

Figure 7.10.: Macro averaged binary precision at standard recall levels for relevance setting Binary 7

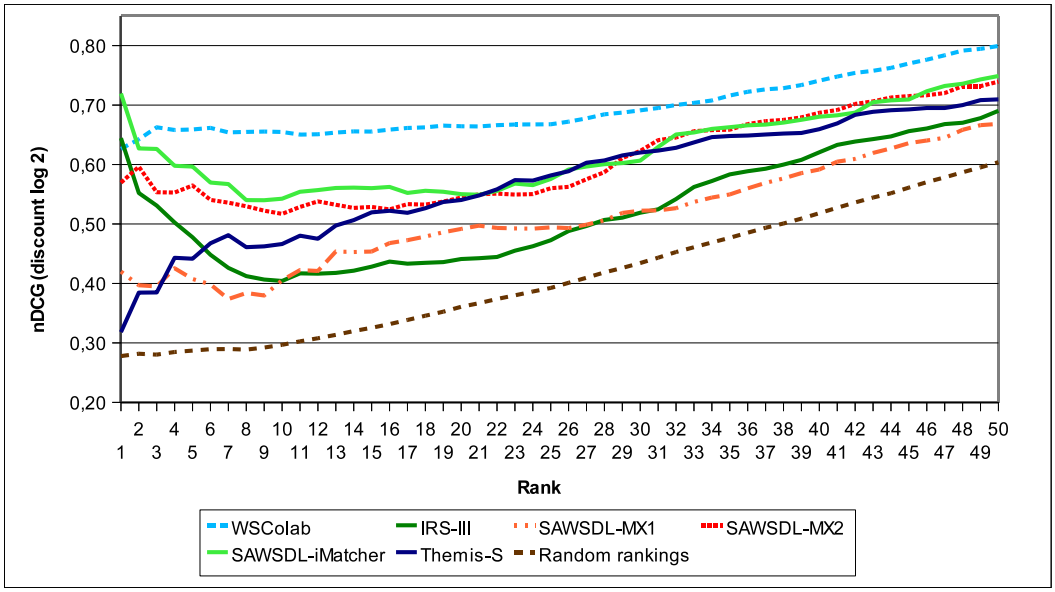

Figure 7.11.: Normalized discounted cumulated gain for discount function $\log_2(i+1)$ with relevance setting Graded 1

untrained matchmakers (and SAWSDL-iMatcher which was trained on SAWSDL-TC, i.e., an unrelated dataset).

Third, to put evaluation results into context, we included the performance of a random ranking in most results. This performance was computed by creating fifty random permutations of the services in the test collection and averaging the scores obtained for those fifty rankings. Comparison with another set of performance scores obtained in the same way showed that fifty permutations is a sufficiently large base for stable random results for this test collection.

Fourth, IRS-III and WSColab were originally designed to return only matching service sorted by decreasing relevance. They were required to return full rankings of all services to allow for meaningful comparison of results. Thus, both matchmakers had to return services they deemed irrelevant in random order. IRS-III performs a rather strict (high precision / low recall) matchmaking. Therefore, it had to return many services in random order. This explains the notable decline in performance after the top ranks.

Themis-S is interesting since it behaves significantly different from all other matchmakers. This is evident in both retrieval performance charts given. As can be seen, Themis-S performs comparatively better for the bottom ranks than for the top ranks. This seems to suggest that it is recall rather than precision oriented.

7.9. Analysis of Evaluation Reliability

After having presented the reference execution of the benchmarking approach, we now analyze its retrieval correctness results. Three factors that may influence the evaluation results are of specific interest in this context: The definition of relevance used, inconsistency in relevance judgments and the choice of evaluation measure. These will be analyzed in turn.

Please note that an analysis of the evaluation results with respect to the primary evaluation question of how to improve the participating matchmakers is beyond the scope of this thesis and can only be performed by the developers of the participating matchmakers. Therefore, we concentrate on the classic question about the reliability of the produced ranking of matchmakers, i.e., whether a measure correctly orders the matchmakers by their retrieval effectiveness and is not influenced by other factors not of interest and under control during the evaluation.

7.9.1. Influence of Relevance

Figure 7.12 illustrates the sensitivity of binary AveP to changes in the relevance definition being used. The chart highlights drastic swaps in the relative performance of the evaluated matchmakers. Using Binary 2, for instance, SAWSDL iMatcher and

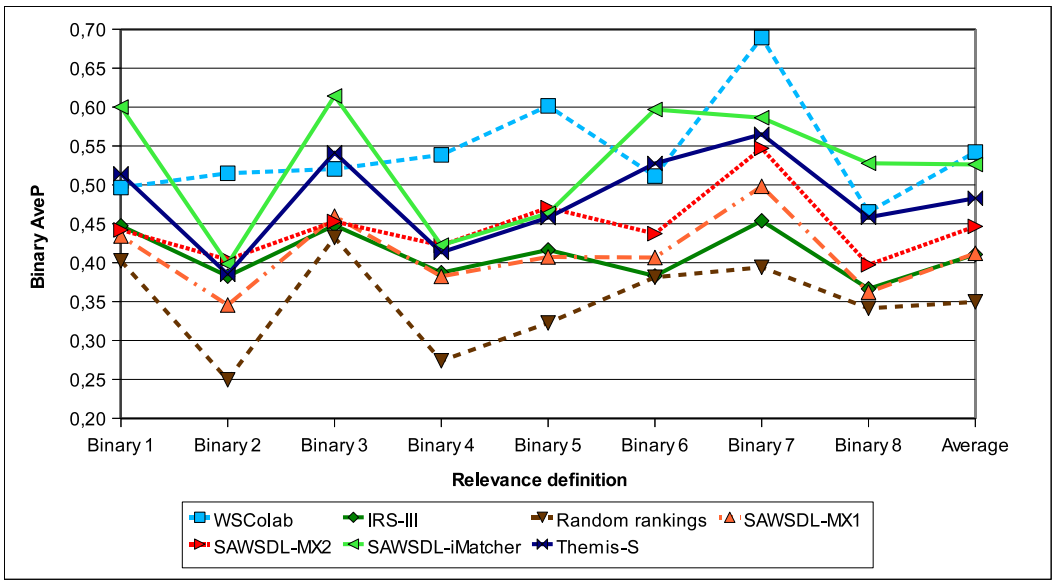

Figure 7.12.: Sensitivity of binary AveP to changes in the relevance definition

Themis-S are significantly inferior to WSColab and roughly as good as SAWSDL-MX2 and IRS-III. In contrast, using Binary 3, SAWSDL iMatcher is clearly superior to all other matchmakers, WSColab and Themis-S are roughly equally good and both significantly better than all remaining matchmakers.

These findings are in line with similar studies from IR that found different retrieval systems to be better at finding highly relevant documents than those being best at finding generally relevant documents, e.g. [Voo01a].

Given that stability of retrieval evaluation results generally decreases with smaller test collection sizes, our results stress that SWS retrieval evaluation results based on the commonly used binary recall and precision need to be taken with a lot of care, since they are highly dependent on the definition of relevance underlying the reference judgments. However, as discussed in Section 7.5, the question of how to properly define relevance in the context of SWS retrieval tends to receive very little attention, thus putting some question marks behind the reliability of the resulting evaluation results.

Figure 7.13 shows the sensitivity of $NDCG_{50}$ with discount $\log_2(i+1)$ to changes in the gain values, i.e., to changes in the relevance definition for graded relevance. As can be seen, the relative performance of the evaluated matchmakers is almost entirely stable. This result is consistent for all retrieval performance measures based on graded relevance. The fact that graded relevance is more stable than binary

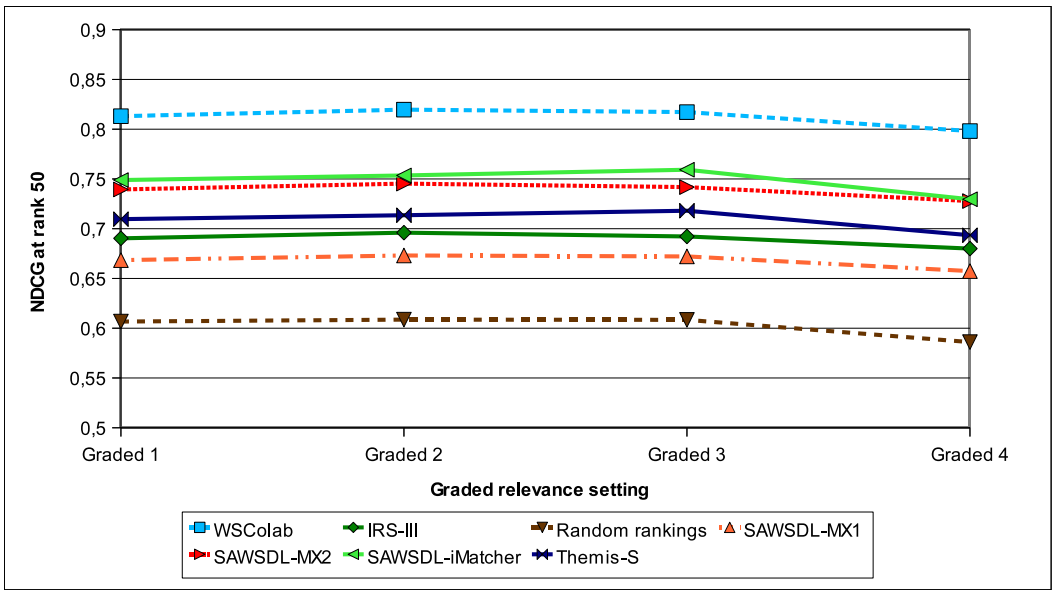

Figure 7.13.: Sensitivity of $NDCG_{50}$ with discount $\log_2(i+1)$ to changes in the gain values

relevance is not surprising. With binary relevance, small changes in the perceived user relevance (as expressed via the reference judgments) may tip a service from being entirely relevant to being entirely irrelevant. Such small changes will only result in moderate changes in the gain values if graded relevance is used, thus leading to much higher stability.

This finding is, again, in line with similar findings from the IR community [Sak07b]. Nevertheless the amount of difference in stability is remarkable. At least our test data makes a very strong case for preferring graded over binary relevance for the given evaluation use case.

7.9.2. Effects of Inconsistent Relevance Judgments

Section 7.6 investigated in depth the issue of inconsistency in reference relevance judgments. We now complement the corresponding discussion by analyzing the effect that judgments by different judges have on the comparative evaluation results. Studies from IR have indicated that inconsistent judgments do not compromise the reliability of IR evaluations, if test collections are large (at least tens of thousands of documents) and results are averaged over many (dozens of) queries [Sar08].

With respect to the questions "Given that relevance judgments are inconsistent, which they are to various degrees as amply demonstrated, how does this affect results of IR evaluation? Because of that, are IR test results valid, reliable and to be trusted in a scientific sense?" Sakai summarizes:

> "In evaluating different IR systems under laboratory conditions, disagreement among judges seems not to affect or affects minimally the results of relative performance among different systems when using average performance over topics or queries. The conclusion of no effect is counter-intuitive, but a small number of experiments bear it out. However, note that the use of average performance affects or even explains this conclusion.
>
> - Rank order of different IR techniques seems to change minimally, if at all, when relevance judgments of different judges, averaged over topics or queries, are applied as test standards.
> - However, swaps – changes in ranking – do occur with a relatively low probability. The conclusion of no effect is not universal.
> - Another however: Rank order of different IR techniques does change when only highly relevant documents are considered – this is another (and significant) exception to the overall conclusion of no effect.
> - Still another however: Performance ranking over individual queries or topics differs significantly depending on the query." [Sar07b, Sar08]

However, since SWS test collections are comparatively small and will remain rather small for the foreseeable future, it is by no means safe to apply this finding to SWS retrieval evaluations and assume that conflicting reference relevance judgments do not compromise the reliability of evaluations in this area, too. In fact, in light of the instability of binary AveP to changes in the relevance definition observed above, one can expect a similar instability of binary AveP to changes of the used reference judge. Figures 7.14 and 7.15 show the computed AveP scores for the Binary 2 and Binary 7 relevance setting using the consensus judgments as well as the original ones obtained from each judge. The figures illustrate that changes in rankings, even notable ones, occur. Nevertheless the influence is much smaller than that of switching the definition of relevance.

We now turn to graded relevance. Figures 7.16 and 7.17 show the computed NDCG_{50} (discount function $\log_2(i+1)$) and ANCG scores for the Graded 1 relevance setting. There were fewer swaps in rankings than with binary relevance, in fact, for NDCG_{50} there was not a single swap for all four graded relevance settings.

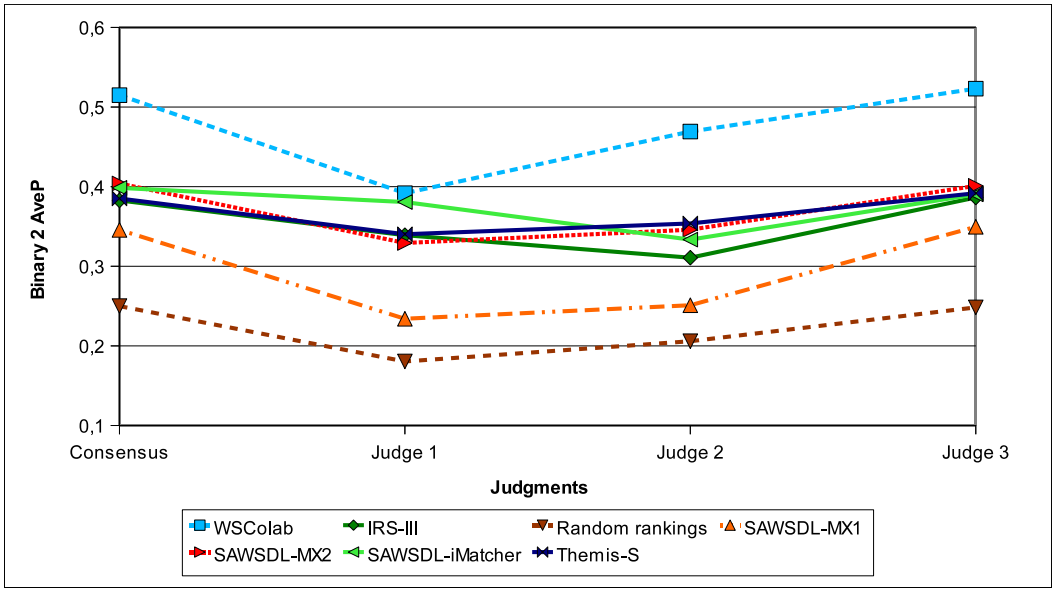

Figure 7.14.: Sensitivity of AveP to inconsistent relevance judgments (Binary 2 relevance)

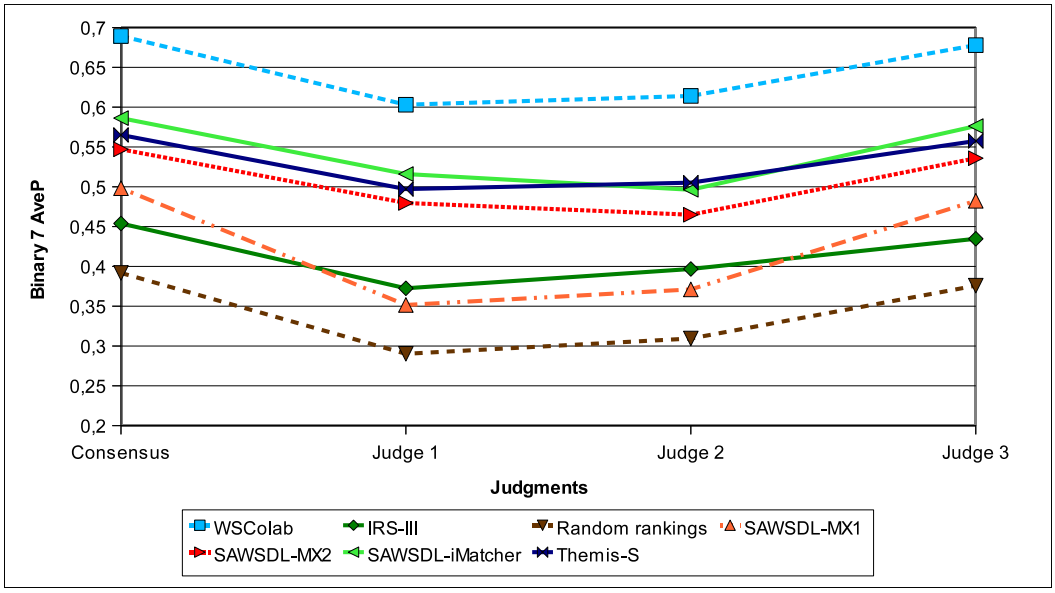

Figure 7.15.: Sensitivity of AveP to inconsistent relevance judgments (Binary 7 relevance)

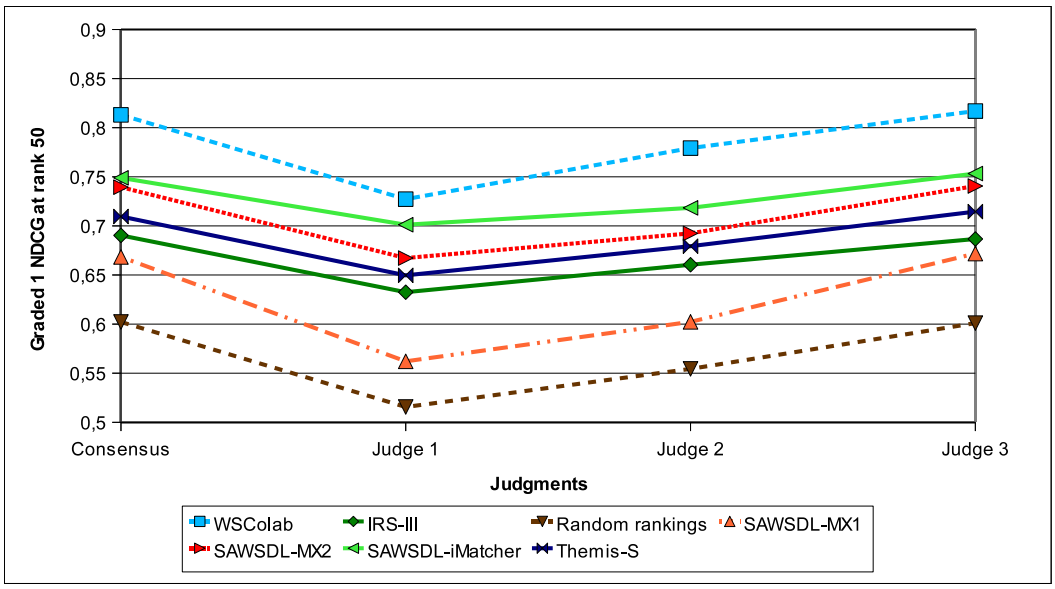

Figure 7.16.: Sensitivity of $NDCG_{50}$ with discount $\log_2(i+1)$ to inconsistent relevance judgments (Graded 1 relevance)

However, for the other measures (ANCG, GenAveP', Q-Measure, ...) a few swaps occurred. In particular IRS-III and SAWSDL-MX1 tended to switch ranks with Judge 1 and Judge 2 favoring IRS-III and Judge 3 and the consensus judgments favoring SAWSDL-MX1.

Concluding, we found graded relevance measures again to be more stable than the binary based AveP. However, with the exception of $NDCG_{50}$, swaps in rankings also occurred using graded relevance. Based on our results, it is difficult to decide whether inconsistent judgments should be considered a serious problem or not. The encountered level of inconsistency in judgments is probably tolerable, at least if graded relevance is used. However, there is still a benefit of having more reliable consensus judgments with respect to the public acceptance of the evaluation and evaluation results, and probably also with respect to their usefulness for an in-depth investigation of the weaknesses of participating matchmakers. Therefore, a case to case decision has to be made about the effort spent for obtaining the reference judgments and the required degree of reliability.

7.9.3. Influence of Evaluation Measure

Finally, we now turn to discussing the influence of the choice of evaluation measure to the evaluation results. We focus on measures based on graded relevance but

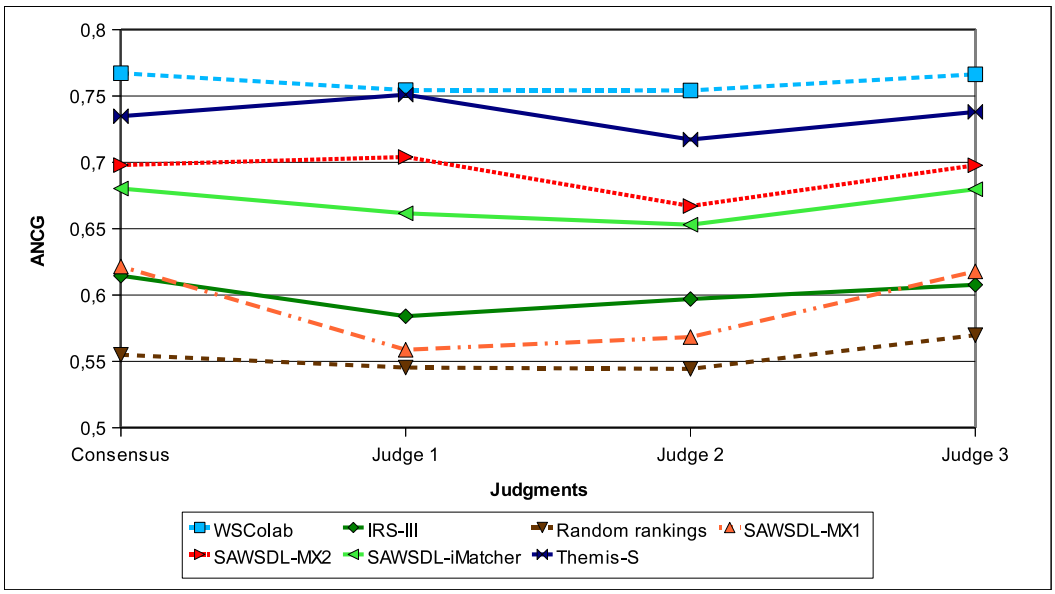

Figure 7.17.: Sensitivity of ANCG to inconsistent relevance judgments (Graded 1 relevance)

include AveP for comparison. However, we do not consider measures based on binary relevance besides AveP. Other binary measures like Precision$_l$ are known to be less stable or have normalization and averaging issues [BYRN99]. Furthermore, all graded retrieval measures can be used with binary relevance judgments anyway. Please note that the AveP included in the following charts is based on a relevance definition that considers all services with a positive gain as binary relevant and is thus different from the binary AveP discussed above. However, using AveP this way allows direct comparison with the graded relevance measures by separating the influence of the measure from the influence of the gain value settings. Furthermore, this is also the AveP version which is blended into Q-Measure.

We consider the following measures: NDCG, ANDCG, AWDP, ANCG, AWP, Q-Measure, AveP, GenAveP and GenAveP'. Q-Measure is used with $\beta \in \{0.5, 1, 2\}$. The measures including a discount are used with the following discounting functions: $\sqrt{i}$, $\log_2(i+1)$, $\log_3(i+2)$ and $\log_5(i+4)$. Figure 7.18 illustrates the discounting behavior of these functions over the relevant ranks 1–50. All measures are evaluated using the Graded 1 gain values (the influence of changing the gain values has been discussed above already).

Note that we included correct as well as incorrect measures. With respect to the correct ones, differences in the evaluation ranking are expected to exclusively reflect

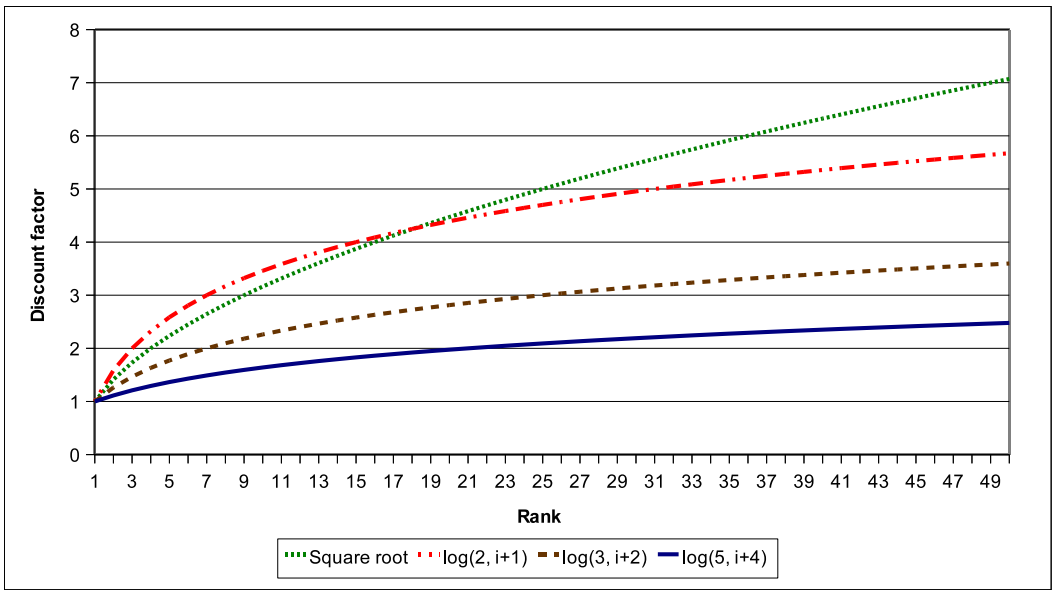

Figure 7.18.: Discount functions used for the comparison of measures

differences in the emphasis of top versus bottom ranks. According to the remarks about the matchmakers made in Section 7.8.5, one can expect that Themis-S benefits from low emphasis on top ranks whereas IRS-III should profit from high emphasis on top ranks.

Figure 7.19 compares ANCG, AWP, Q-Measure, AveP, GenAveP' and GenAveP. The figure illustrates nicely the characteristic of Q-Measure being a blended ratio between AWP and AveP. With respect to matchmaker order, swaps occur, even though exclusively among Themis-S, SAWSDL-MX2 and SAWSDL iMatcher.

However, the figure highlights a quite drastic difference in measure behavior between the incorrect AWP and its fixed counterpart ANCG. Please recall that AWP had two defects. First, its inability to punish very late retrieval, second, its property of rewarding correct order of relevant items rather than their absolute ranks. Themis-S is the only matchmaker whose score declines when switching from ANCG to AWP. This can be explained through the first AWP defect, since Themis-S, as discussed above, is inferior at retrieving highly relevant items at the top ranks, but superior at retrieving all relevant items relatively soon. The first characteristic is correctly punished by both measures, whereas the second is not rewarded by AWP.

By comparing the correct ANCG and GenAveP' with the incorrect AWP and GenAveP, one realizes that Themis-S also suffers from the second AWP defect. Please recall that GenAveP shares the order versus rank defect with AWP, but not

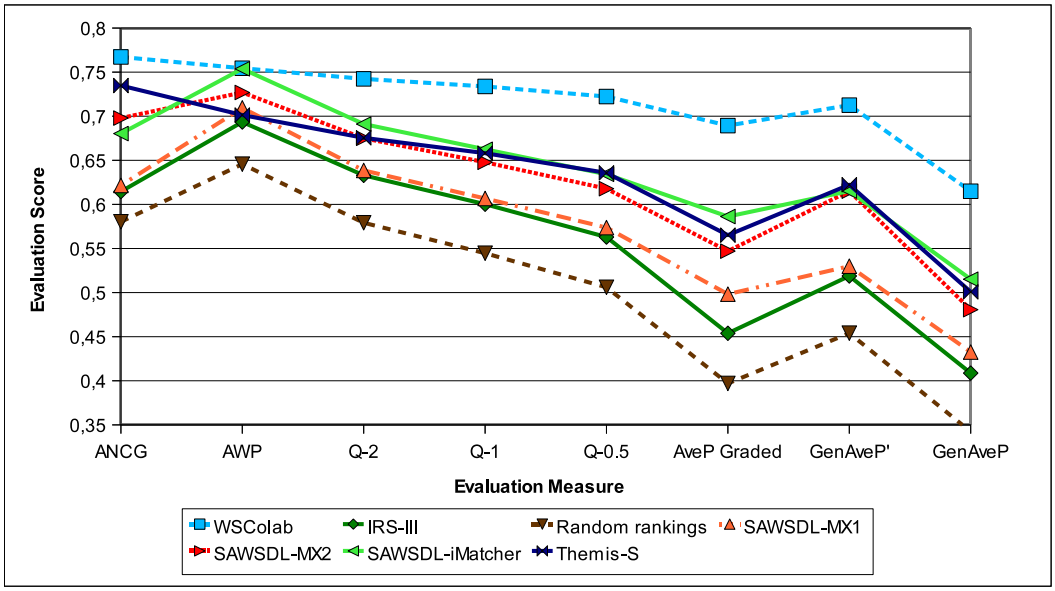

Figure 7.19.: Comparison of graded evaluation measures (Part 1)

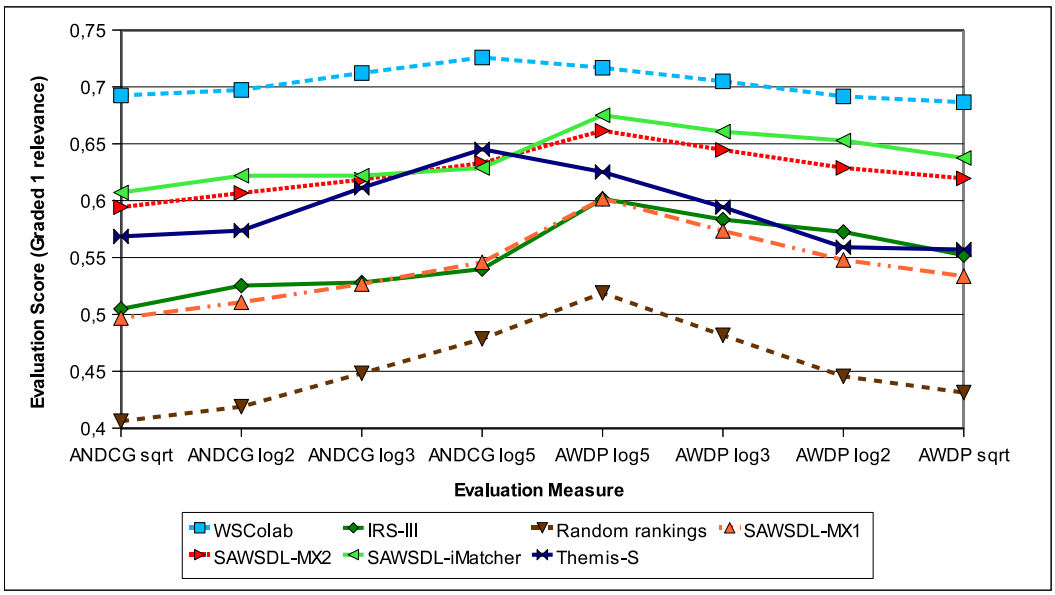

Figure 7.20.: Comparison of graded evaluation measures (Part 2)

AWP's inability of properly punishing late retrieval. This probably explains the smaller differences in relative performance between GenAveP and GenAveP' while the – even though narrow – swaps in matchmaker order that still occur between these two measures is likely caused by the order versus rank defect which GenAveP does suffer from while GenAveP' does not.

Figure 7.20 compares the incorrect AWDP with its correct counterpart ANDCG using different discount functions (please note that $NDCG_{50}$ rated equal to ANDCG and was thus not included in the charts). Comparing the relative scores of Themis-S once more highlights the effect of the order versus rank defect that AWDP suffers from. Besides, this figure nicely illustrates the effect of discounting. Stronger discounting comparatively benefits IRS-III whereas Themis-S profits from smaller discounts. This was entirely expected and results from the retrieval behavior of these matchmakers discussed above.

7.9.4. Conclusions

Some important conclusions for future retrieval effectiveness evaluations may be drawn from the analysis provided above.

1. Binary AveP is highly sensitive towards changes in the definition of relevance underlying the reference judgments. Unless one knows about this definition very well, is certain that the definition matches the use case of the evaluation and that the reference judges know to apply the definition correctly, we recommend against using binary relevance in the future.

2. Graded relevance is extremely stable against moderate changes in the gain values (and thus the underlying definition of relevance).

3. Inconsistency in reference judgments influences evaluation results, but only moderately. Again, binary relevance is less stable than graded relevance. Obviously, more reliable judgments are preferable, but the effects of inconsistency seem to remain in a tolerable range, at least for graded relevance.

4. The choice of evaluation measure influences the evaluation results. The choice of a graded measure has less influence than the choice of relevance with binary AveP, but more influence than inconsistent judgments. One should not choose a particular measure without reflecting why this measure is chosen. In case of doubt, it seems wise to analyze with different measures and report correspondingly. Contradicting measures also indicate significantly differing retrieval characteristics of the matchmakers exchanging ranks and thus allow tracing those characteristics.

5. As was suggested before, AWP is not a reliable evaluation measure because of its inability to properly punish late retrieval. However, by rewarding order of relevant items rather than their absolute ranks, Q-Measure, AWDP and GenAveP may also show an unintuitive measure behavior. AWDP suffers from this problem most, while Q-Measure and GenAveP suffer significantly less.

6. ANCG/ANDCG and $NDCG_l$ are correct and offer the most intuitive and flexible way of customizing the emphasis on top over bottom ranks. These measures seem to be preferable for retrieval effectiveness evaluations. NDCG charts are probably the most informative way of presenting evaluation results, since they provide an indication of the performance of matchmakers over ranks and still provide a summary measure by the value at the bottom rank ($NDCG_{50}$ in our case).

Please note that, as usual with experimental results, these conclusions are not universal. The experiment that the analysis is based upon was relatively small in terms of number of services and matchmakers and it is impossible to eliminate all potential effects of particular data or matchmaker characteristics. With this respect, we would like to quote Saracevic from a meta-study on the effects of relevance to IR evaluations:

> "Caveats abound. Numerous aspects of the studies reviewed can be questioned and criticized. Criteria, language, measures, and methods used in these studies were not standardized and they varied widely. [...] Still, it is really refreshing to see conclusions made based on data, rather than on the basis of examples, anecdotes, authorities, or contemplation. [...] As mentioned, generalizations should primarily be treated as hypotheses." [Sar07b]

7.10. Related Work

In the following, we provide a brief discussion of the work directly related to the benchmarking approach presented in this section, structured by work from the area of SWS retrieval, web service retrieval and general IR.

7.10.1. SWS Retrieval

Experimental evaluation of SWS matchmaking and retrieval has received very little attention so far [KKRPK08]. Most approaches to SWS discovery and matchmaking are presented without being experimentally evaluated and rather illustrated by examples, e.g., [RSN+07, Lar06, JRGL+05, GMP06].

Of the remaining ones, quite a few make rather controversial assumptions on the evaluation setting without justifying them properly. An extremely common approach, for instance, is to evaluate a matchmaking algorithm based on semantic formalizations which are simply assumed to be complete and correct: "Since our matchmaking process will return only the web services that can satisfy all capabilities requested in the query [...] we assume precision is 1 as all services returned are definitely relevant to the query" [SS06]. Obviously, this almost reduces the evaluation of a matchmaking algorithm to a discussion about the correctness of its implementation.

Few approaches present a rather thorough evaluation (e.g., [KKF08, KK07, NSDM03, BOI09, VHA05, ÅÅLS06]). Unfortunately, not all of them make their evaluation data public (in fact, of the given examples only [KKF08]), thus rendering a comparative evaluation or an informed analysis of the reported evaluation results infeasible.

Furthermore, evaluation approaches in the area of SWS retrieval so far adopted the standard evaluation methodology from IR, i.e., they relied exclusively on binary relevance and standard measures based on precision and recall. A discussing of the suitability of this approach for the special case of semantic service retrieval is generally lacking. This was also the case with the first editions of the S3 Contest on Semantic Service Selection. However, within the context of this thesis the contest has started to integrate alternative measures based on graded relevance in 2009.

We are not aware of any previous work specifically on evaluation methodologies for SWS retrieval except for the work by Tsetsos et al. [TAH06]. They were the first to propose the usage of a graded relevance scheme for SWS evaluation, but their scheme (irrelevant, slightly relevant, somewhat relevant, relevant, very relevant) lacks precise definitions of the relevance levels. They dealt extensively with graded versus binary relevance and experimented with a subset of OWLS-TC. However, different behavior of evaluation measures, new test collections or the consistency of relevance judgments and its consequences were not in the focus of their work.

With respect to measures, they propose to use a relevance scale based on fuzzy linguistic variables and the application of a fuzzy generalization of recall and precision that evaluates the degree of correspondence between the rating (not ranking) of a service by an expert and a system under evaluation. In this aspect this measure is very similar to the ADM (average distance measure) measure proposed by [MDGM06]. Unlike measures that evaluate the ranking created by a retrieval system these measures evaluate the absolute relevance score assigned to a retrieved item by the system. This can lead to counterintuitive results since such measures are obviously biased against systems that rank services correctly but generally assign relatively higher or lower scores [Sak04a]. The measures that we use in this chapter avoid this issue.

Apart of the work by Tsetsos et al., we are not aware of any work in the area of semantic services that explicitly targets evaluation methodologies, including relevance models, relevance judgments or evaluation measures.

With respect to reference judgments, Di Noia et al. obtained reference rankings for service matchmaking evaluations by directly asking human assessors to rank the available services [NSD07]. This approach avoids the imprecision related to binary relevance judgments and generally yields more stable results than inducing a reference ranking via relevance judgments. However, it also requires much more effort from the human assessors and is thus difficult to scale to larger datasets. Besides, it does not allow the emulation of different relevance models with one set of judgments. This capability is an advantage of the multi-dimensional relevance model we propose.

Di Noia et al. evaluate matchmaking performance using rank correlation measures from statistics. As discussed above, these measures estimate the difference between two rankings but, for instance, do not differentiate whether the rankings differ in the top ranks or the bottom ranks.

7.10.2. Web Service Retrieval

The work in this chapter is not only applicable to semantic service discovery but also directly applies to more traditional WSDL matchmakers (in fact, Themis-S can be considered as one of these). Among others, Dong et al. [DHM+04], Stroulia and Wang [SW05], and Kokash et al. [KvdHD06b] have proposed such matchmakers.

For evaluation, Dong et al. used a dataset of 431 Web Services (WSDLs) retrieved from the Web. Unfortunately, the dataset or any information on the applied relevance definition (beyond "similar operations") is not available.

Stroulia and Wang reused an existing dataset of 814 WSDL description originally created by Kushmerick and Hess[14]. Kushmerick and Hess classified the services hierarchically into seventy categories and Stroulia and Wang used that classification as relevance judgments, i.e. services belonging to the same category were considered to be relevant to each other, all others were considered irrelevant. This is a very broad definition of relevance that is not usable for SWS retrieval evaluation.

Kokash et al. also reused that collection and additionally built their own collection of 40 services classified in five categories. They used the same broad definition of relevance as Stroulia and Wang. None of the mentioned publications deals with relevance or relevance judgment consistency in particular or test collection construction and evaluation methodologies in general in the context of service retrieval.

[14] `http://www.andreas-hess.info/projects/annotator/`

7.10.3. Information Retrieval

As evaluation of SWS retrieval generally should, the work presented in this chapter follows the giant footprints of IR. Evaluation methodologies and the construction of test collections have been extensively studied in the IR community over decades. A complete coverage is available in the IR standard literature (e.g., [BYRN99]) and only the most relevant references are provided here.

Saracevic was already cited frequently throughout this chapter. He has published extensively about IR evaluation in general [Sar95], relevance and relevance judgments as well as their importance for IR evaluation [Sar07b, Sar07a] or the effects of inconsistency of relevance judgments on IR test results [Sar08].

Tague-Sutcliffe, Sparck Jones and Voorhees are mentioned as other examples for extensive work dealing with IR evaluation in general [TS92, Sar95, Voo01b, Voo01a, Jon00, Jon05].

Kekäläinen, Järvelin, Sakai and Kishida have published on the characteristics of metrics based on graded relevance, e.g. [Sak04a, KJ02, JK02, Sak07a, Sak07b, Sak04b, Kis05].

We rely heavily on all these achievements and our work can be viewed as an application and adaptation of this work to the SWS retrieval evaluation domain. We are not aware of any previous work on relevance schemes specifically designed for the SWS retrieval domain and discussions on how to provide reliable and consistent relevance judgments within this domain. Furthermore, we are not aware of a systematic discussion of the properties of all the measures covered in this chapter, in particular not with respect to what we defined as correctness. To the best of our knowledge, the order versus rank problem in measures like Q-Measure, AWDP or GenAveP has not been discussed previously.

7.11. Summary

This chapter presented a benchmark for evaluating SWS discovery and matchmaking approaches with a specific focus on retrieval correctness. The primary goals of the benchmark are to provide means to reliably assess the comparative strengths and weaknesses of current approaches, to investigate the trade-off between approaches using different levels of semantic formalizations and to provide comparative data to help matchmaker developers improving their techniques.

The novel contributions of the chapter are as follows. A methodology for the comparative evaluation of SWS matchmakers across formalisms was presented. This methodology comprised a discussion of the differences between SWS and general information retrieval and its effects to the design of suitable evaluation setups. A corresponding evaluation setup was presented and its theoretic background was thoroughly investigated. This included the development of a relevance model for

SWS retrieval, an in-depth investigation of issues around relevance judgments for SWS retrieval evaluation and a discussion of the properties of various retrieval correctness evaluation measures. Improvements for measures from IR were proposed to resolve identified shortcomings of the existing measures.

Based upon this theoretic discussion, a reference implementation of the benchmarking approach was presented. The reference implementation was organized as a community benchmarking event under the umbrella of the well-established S3 Contest on Semantic Service Selection. It provided the first opportunity ever to investigate the comparative performance of service matchmakers relying on entirely different formalisms.

Using data retrieved from this benchmarking event, an analysis of the benchmarking approach was provided. This analysis comprised a discussion of the effects of the choice of relevance, inconsistencies in reference relevance judgments and the choice of evaluation measure to the evaluation results. General recommendations for future evaluations in the area were derived from this analysis. Finally, related work relevant to this chapter was discussed.

This chapter concludes the main part of this thesis. A critical appraisal of the contributions of the thesis, including a review of the benchmark presented in this section, will be provided in the following Chapter 8.

Part III.

Finale

CHAPTER 8

Validation

> We really haven't got any great amount of data on the subject, and without data how can we reach any definite conclusions?
>
> *(Thomas A. Edison)*

In this chapter, the contributions from Part II are validated. First, Section 8.1 provides references to validations already reported throughout this thesis. Afterwards, Section 8.2 introduces the validation approach that was applied to evaluate the thesis. Subsequently, Section 8.3 validates the assumptions which the thesis and its solution approach are based upon. Following, Sections 8.4 through 8.7 validate the contributions of the thesis from the corresponding Chapters 4 through 7. Finally, Section 8.8 provides a summary of the validation results.

8.1. Already Reported Validations

Various means of evaluation have already been taken within Part II and will not be repeated in this chapter. Instead, we provide pointers to the corresponding sections for reference before we start with the complementary validation of the thesis in the following section.

Some data and usage statistics of the OPOSSum portal have been reported in Section 5.3.3. The applicability of the portal is also evidenced by the integration of existing collections into the portal. This integration as well as some discussion of the experiences gained have been reported in Section 5.3.4. The contributed Jena Geography Dataset is discussed and compared with existing collections in Section 5.4.

The methodology for benchmarking the functional scope of SWS frameworks has been developed in cooperation with a number of groups also active in the SWS Challenge initiative. Parts of it have also been developed in a W3C incubator group. Thus, these contributions have undergone extensive reviewing by other scientists. The relationship between the contribution and the SWS Challenge initiative is clarified in Section 6.1. The concrete scenarios have also been assessed and approved by the SWS Challenge organizing committee as well as an evaluation workshop of the initiative. This is described in Section 6.3.1.

The setup and methodology for benchmarking SWS matchmaking has been assessed and approved by the S3 Contest organizing committee when the benchmark was executed as part of the 2009 edition of the S3 Contest. This has been reported in Section Section 7.8. Furthermore, an in-depth comprehensive evaluation of the relevance judgments and metrics used by this benchmark has been presented in Sections 7.6, 7.7.4, 7.7.5 and 7.9.

8.2. Validation Approach

The approach we take towards evaluating the contributions of this thesis involves three aspects. As a prerequisite for the evaluation of the thesis contributions, we perform a critical appraisal of the emphasis on community involvement to SWS technology evaluation in the following Section 8.3. We first validate the assumption that SWS related research is ready for community-based benchmarking. This will be followed by a discussion of the lessons learned and experience gained during the process of organizing and executing community-based benchmarking campaigns.

Subsequently, we perform an evaluation of the thesis contributions. This evaluation aims at verifying or falsifying whether the engineering objectives of this thesis have been achieved and whether this has been done in compliance with the specified requirements. We thus repeat the thesis objectives and requirements from Section 1.3 and discuss how these will be verified. The objectives of this thesis were twofold:

Objective 1: Development of a comprehensive and well-founded conceptual model for SWS technology evaluation.

- **O1.1:** Identify evaluation dimensions, i.e., the criteria to evaluate.
- **O1.2:** Identify evaluation requirements to promote and ensure evaluation quality.

Objective 2: Provide reference benchmarks for selected evaluation criteria and use cases to solve concrete benchmarking needs in the area.

- **O2.1:** Identify measures for selected criteria.

O2.2: Design and implement measuring instruments to assess a system with respect to these measures.

O2.3: Develop and establish methodologies to obtain measurements and conduct an evaluation.

These objectives had to be achieved in compliance with the following requirements:

Main Requirements: Impartiality, community participation, continuous application.

R1: Ensure the applicability of the evaluation framework and the developed benchmarks to different technologies and avoid unnecessary prerequisites and any biases to particular approaches.

R2: Promote a culture of collaboration for enabling community input and feedback during the process of benchmark development.

R3: Establish structures to foster the dissemination of the benchmark and the continuous co-evolution of the benchmarking efforts and the scientific community.

O1 has been approached through the conceptual framework presented in Chapter 4. The framework comprises a criteria model which identifies the dimensions of evaluation (O1.1) as well as a requirements catalogue to promote and ensure evaluation quality (O1.2). Both will be validated in Section 8.4.

O2 has been approached through the contributions presented in Chapters 5, 6 and 7. Enabling the collaborative development of large scale SWS test collections lays foundation for the implementation of many SWS evaluation measuring instruments (O2.2). The corresponding contribution is evaluated by means of its prototypical implementation, i.e., the OPOSSum portal and the Jena Geography Collection developed within this portal. Both evaluations will be presented in Section 8.5.

Furthermore, two benchmarks have been developed within this thesis (O2). Both comprise measures (O2.1), measuring instruments (O2.2) and a methodology for obtaining measurements and conducting the actual benchmark execution (O2.3). The corresponding meta-evaluation is provided in Sections 8.6 and 8.7. Each benchmark has been executed within a community-wide benchmarking campaign, i.e., it has been implemented and executed in practice. Thus, the suitability and effectiveness of the benchmarks can and will be assessed by discussing these reference benchmark executions. Additionally, the benchmarks will be assessed against the requirements catalogue for evaluations developed as part of the conceptual framework. This assessment serves as an extended validation of R1. Finally, R2 and R3 will be assessed through a discussion of the dissemination activities performed as part of the benchmark executions.

8.3. Discussion of Thesis Approach

In this section, we validate the assumption that SWS related research is ready for community-based benchmarking and provide a critical appraisal of the community-based benchmarking approach that this thesis followed.

Sim identifies three conditions that must exist in a scientific research community before construction of a benchmark (as a means for improving scientific results and consensus in a discipline) can be fruitfully attempted. These are

- a necessary minimum level of maturity
- a tradition of comparing research results and
- an ethos of collaboration.

They will be briefly discussed in turn, but for an extended discussion we refer the interested reader to [Sim03].

Minimum Level of Maturity: This is important because of the downsides potentially involved in benchmarking. First, the development and maintenance of a benchmark involves significant cost. Without a minimum level of technological maturity, the resources invested in the development of a benchmark will be wasted. Second, there is a danger in committing to a benchmark too early. Sim remarks: "Locking into an inappropriate benchmark too early, using provisional results, can hold back later progress. The advantage of having a benchmark is that the community works together in one direction. However, this commitment means closing off other directions, albeit temporarily" [Sim03].

Tradition of Comparison: This is important for a benchmark being successful. First, it indicates that the community recognizes the importance of validation and thus of the benchmark. Without this, any benchmarking attempt will have little impact. Second, there is a body of research that the benchmark development can start from which eases the process or even makes it feasible in the first place.

Ethos of Collaboration: This, too, is important for a benchmarking attempt being successful. Without a willingness to work together to solve common problems, a community will not join forces to develop a benchmark towards becoming a standard. An ethos of collaboration, a familiarity and experience with working together indicates a community that is receptive to the results of a benchmark and thus more likely to use the benchmark.

	Maturity	Comparison	Collaboration
SWS technology benchmarking readiness score	10.5	6	5
Interpretation of Scores			
Too Soon	0–4	0–4	0–3
Ready for Benchmarking	5–9	5–9	4–7
What are you waiting for?	10–12	10–12	8–10

Table 8.1.: SWS technology benchmarking readiness assessment results

8.3.1. Assessment of Community Benchmarking Readiness

Sim provides questionnaires for assessing the readiness of a community to commit to benchmarking [Sim03]. The questionnaires comprise a number of multiple choice questions regarding the maturity, the standards of comparison and the ethos of collaboration. Each answer is awarded points (0 points for each "a", 1 point for each "b" and 2 points for each "c") and a resulting score is computed for each prerequisite. The answered questionnaires are provided in Tables 8.2, 8.3 and 8.4. The scores and their interpretation are displayed in Table 8.1.

According to this assessment, SWS research is ready for benchmarking regarding all three prerequisites. Regarding maturity it is even assessed the highest level of readiness. Thre tradition of comparison and collaboration are assessed somewhat weaker. This corresponds to the discussion provided in Chapter 3 which already highlighted the previous lack of comparison among alternative SWS approaches.

The assessment of the benchmarking readiness according to the methodology developed by Sim will be complemented in the following section by an in-depth discussion of the experiences and lessons learned while organizing and executing the benchmarking campaigns as part of this thesis work.

8.3.2. Experiences and Lessons Learned

Community based benchmarking can be opposed to unilateral evaluations. The latter involve a comparison of different technologies by a single interested party. In contrast, community based benchmarking is based upon establishing common benchmarks and encouraging researchers to participate in community evaluation initiative such that the developers of the evaluated technologies are closely involved in the evaluation.

Subsequently, we discuss some strengths and weaknesses of the community based approach by means of experiences and lessons learned. This covers the involvement of developers in the evaluation, the effort necessary by organizers and participants

Maturity	Response
How many years ago did this research area split from another one? a) four or fewer b) five to ten c) ten or more	b) The area was defined almost ten years ago ([MSZ01]).
How many implementations are there of technology under study? a) two or fewer b) three to five c) six or more	c) There are more than six implementations of SWS technology algorithms (see [Klu08a]).
What phase of maturity has the technology reached on the Redwine and Riddle Maturity Model? a) Basic Research, Concept Formulation, Development and Extension b) Enhancement and Exploration (internal) c) Enhancement and Exploration (external) or Popularization	b–c) The technology shows characteristics of internal and external Enhancement and Exploration.
How many annual conferences and workshops are dedicated to this research area? a) none b) one or two c) three or more	c) There are several conferences with special tracks on SWS (ESWC, ISWC, ICSC, ICSOC, ECOWS, ICWS, WWW, ...).
How many journals are dedicated to this research area? a) none b) one c) two or more	c) There are several journals that publish regularly papers from SWS research (IJWS, IJSWIS, IJWGS, IJWSP, IJSC, IJEC, ...).
How difficult is it to publish a speculative paper in one of the above meetings or journals? a) not applicable or easy b) somewhat difficult c) very difficult	c) It is very difficult to publish a speculative paper in one of the above listed meetings or journals.

Table 8.2.: Benchmarking Readiness Assessment — Maturity

Comparison	Response
How difficult is it to publish a paper that introduces a new technology without validation? a) easy b) somewhat difficult c) very difficult	b) It used to be easy but it gets increasingly difficult.
How many different implementations of the technology have been applied to solve an industrial problem? a) two or fewer b) three to five c) six or more	b) According to project deliverables, several implementations have been applied to solve industrial problems within the large EU FP6 projects.
When was a paper that compared three or more approaches or implementations last published? a) never or more than five years ago b) one to four years ago c) within the last year	b) Detailed comparisons of technologies were published in 2008 [PLZM08].
Does the research area use standard proto-benchmarks or benchmarks to compare technology? a) no b) proto-benchmarks c) benchmarks	b) Proto-benchmarks are in use within the community based evaluation initiatives in the area.
Have there been attempts to replicate the results of these comparisons? a) no b) using the same technology c) using the same technology and evaluation method	a) We are not aware of any replications of the evaluation results.
Have there been tutorials or workshops on how to conduct empirical studies in this particular research area? a) no b) conference session to half day c) full day or longer	c) There have been several multi day workshops on the topic within the SWS Challenge initiative.

Table 8.3.: Benchmarking Readiness Assessment — Comparison

Collaboration	Response
Is time set aside at conferences and workshops for discussions? a) no b) once or twice per meeting c) almost every session of the meeting	b) Time for discussion is regularly scheduled.
Has there been a seminar or workshop in this research area dedicated to discussion and interaction, e.g., a Dagstuhl seminar? a) no b) once c) regularly	b) The SWS Challenge initiative schedules regularly workshops dedicated to this end, but these do not necessarily represent the whole community.
How often do research groups meet to exchange ideas, tools, or techniques? a) rarely b) occasionally, but meetings are not repeated c) regularly (once per year or more)	b) There are regular meetings of the community (e.g., within the SWS Challenge), but they do not cover the whole community.
Have there been many multi-site, multi-year research projects, consortiums, or task forces? a) no b) one or two c) three or more	c) There have been numerous multi-site, multi-year research projects, e.g., within EU FP6.
Are there community-wide standards for paper formats, data exchange, or auditing of research results? a) no b) one c) two or more	a) We are not aware of such standards.

Table 8.4.: Benchmarking Readiness Assessment — Collaboration

and the central issue of how to motivate participation in a community based benchmarking initiative.

Developer Involvement in the Evaluation

Involving technology and tool developers in the evaluation has proven to be quite crucial at the current stage of affairs. Both within the SWS Challenge and the S3 Contest most participating tools have suffered from severe technical difficulties which could not have been overcome without the help of the tool developers. In fact, several tools were specifically developed or at least extended for participation in the evaluation campaigns. This alone requires frequent interaction between the tool developers and the evaluation organizers.

In particular within the S3 Contest, where tools need to be installed on the evaluation organizer's machines, it became evident that this is currently largely infeasible without the help of the tool developers. The tools' maturity was usually insufficient to allow for an out of the box installation of the software.

Furthermore, within the SWS Challenge, it turned out that many solutions to the problem scenarios were not entirely straightforward in all aspects, but instead involved workarounds for some problem aspects particularly challenging to each approach. Usually, in such cases the evaluation results have been augmented with caveats in form of footnotes that highlight and explain the applied workarounds without intending to imply any value judgment. It can be assumed that people not very intimately knowledgeable about the evaluated technologies would not have devised these workarounds.

An important experience is that it is extremely difficult to objectively judge whether a shortcoming of an evaluated technology is fundamental or simply due to a weak implementation status and lacking tool support, problems rather common and natural in academic settings, at least for a field as young as SWS research. The problem is aggravated by the fact that proposed technologies are often implemented prototypically and thus rather poorly documented and furthermore in a state of ongoing development. People not intimately involved in and knowledgeable about the evaluated technologies would probably have failed to use them in an optimal way, at least when it comes to the mentioned workarounds quite common in the solutions so far.

Thus, involving developers in the evaluation ensures a level playing field where all participating technologies are evaluated based upon a comparable level of familiarity and no technology is discriminated because its proper usage may require additional training. This also greatly promoted the acceptance of the evaluation results by the tool developers.

Naturally, the community based approach did not come without disadvantages For one thing, organizing evaluation as a community event involves significant com-

munication and organization overhead. This will be discussed in the following sections. For another thing, an evaluation approach where tool developers use their own tools would not be suitable, if the usability or ease of learning of a technology is the evaluation criteria of primary interest. This, however, was not the case with the benchmarking efforts performed as part of this thesis.

Effort by Organizers

The effort required for the organization of a benchmarking campaign was generally largely underestimated. Even apart from the actual assembly and development of the benchmark, the organization of the evaluation event involves significant communication effort. This was particularly evident in the SWS Challenge shortly prior to each workshop and in the S3 Contest when problems during the installation of the evaluated tools on the organizer's machines had to be overcome. Also the preparation of the evaluation report for the S3 Contest required several weeks of effort including additional communication with selected participants to resolve ambiguities or disagreements with the presentation of results.

Even more effort was involved in case of the SWS Challenge for maintaining a testbed of actually working web services. However, at least during the first years of the challenge, much of the corresponding effort resulted from partially premature web service technology. This has improved significantly since then. Nevertheless, programming and organizing the testbed is still very labor intensive. The involved effort is also the primary reason why the correctness verification of solutions has not been automated to the greatest possible extent so far. Instead, the challenge largely still relies on manual verification resulting in increased manual effort and a decreased ability to easily reproduce the evaluation results.

It also turned out that participants crucially needed support by the organizers to debug their solutions. Web services involve a large layered stack of complex technologies, including web servers, servlet containers, SOAP engines, possibly workflow engines, XML libraries etc. on top of the actual service implementation, which in turn may consist of an entire complex stack of technologies. It turned out that errors were often not properly propagated through the technology stack resulting in meaningful error messages getting lost on the way through the stack. Basically, debugging of even trivial errors proved largely impossible without access to the various server logs. Therefore, continuous support by the testbed maintainers was required, especially prior to workshops when participants were actively developing their solutions. The situation could be improved but not entirely solved by providing participants with automatically updated views on various aspects of the server state and different log files. On the other hand, the creation and maintenance of these views obviously further increased the effort involved in the development of the testbed.

Overall, the level of effort required by the benchmark organizers presents a continuous struggle for both evaluation campaigns and probably represents the primary Achilles heel of both benchmarking approaches.

Effort by Participants

Participants, including the author, probably most drastically underestimated the effort involved in advancing their implementations to a state sufficient for participation in the benchmarking. This was particularly true for the more complex scenarios that had to be implemented as part of the SWS Challenge benchmarking. The effort of creating working, implemented solutions even for problem scenarios that did not appear overly complex at the first glance was substantial. This raises some skepticism with respect to largely unverified claims of most papers published in this field:

> "This Challenge has exposed the fact that academic claims of being able to solve problems should be viewed very critically until they are verified by a methodology similar to that of the Challenge. Every participant has found that solving even the simplest Challenge problem has been far more difficult than anticipated, no one has solved all of the problems, and at least one participant worked for an extended period of time without solving a single problem. At least one semantically-oriented team has not attempted to solve other problems after seeing the effort required to solve the first 'simple' problem." [PKM+08]

The encountered difficulties and the resulting high effort were primarily caused by two factors. First, it appears that the proof of concept implementations of academic results in the area are still rather far from industrial strength, even though often claimed otherwise. This is not necessarily as problematic as it sounds, since the focus of academic work is on the research and development, not the implementation of technologies. Nevertheless the community might need to spend more effort on the implementation of the proposed technologies. The lack of readily downloadable tools und easily usable implementations as a general barrier to the advancement of the field and the transition of results to industry has been mentioned previously already.

Second, apparently approaches being developed on some use cases often do not transfer smoothly to other related use cases. One of the lessons learned from the SWS Challenge is that people are too focused on too few use cases. Confronting the proposed technologies with other — even closely related — scenarios that have been specified outside of a specific project context and thus had not been considered during the development of the evaluated technologies often results in problems. This is a more general issue that highlights the necessity of independent evaluations based

upon common benchmark problems that are not reverse engineered from existing solutions.

On a more general level, the substantial effort involved in the participation of the benchmarking also resulted from the fact that there are no standard formalisms and interfaces against which everyone can be evaluated. If such standards existed, effort for participants could be reduced by providing more shared materials as part of the benchmark.

Providing formalizations of the problem scenarios of the Functional Scope Benchmark, for instance, would reduce the effort for participants working with the corresponding formalism, but also disadvantage or even exclude other participants. This will be further discussed in Section 8.6. Similarly, the first two tracks of the S3 Contest (OWL-S and SAWSDL matchmaking evaluation, see Section 3.1) presuppose a standard interface and a formalism to use. This reduces the effort of participation significantly, but at the cost of also reducing the significance of the evaluation results, as discussed in Section 7.3.

Generally, community based benchmarking has two opposed effects on the effort involved in evaluations. On the one hand, the existence or creation of standard benchmarks relieves people from designing and implementing evaluations from scratch. Benchmarks are meant for reuse and thus save people a lot of effort and actually often make proper evaluations feasible in the first place. The SWS Matchmaking Benchmark, for instance, made evaluations possible that were largely infeasible before.

On the other hand, having standards that people need to adhere to typically also results in additional effort. This additional effort can be well justified and worthwhile but also simply represent unwanted ballast. Both kinds may be illustrated by examples from the Functional Scope Benchmark.

An example of the first kind is the effort people had to spend to work on shortcomings of their technologies they otherwise probably would have circumvent by adjusting the problem definition to their technology instead vice versa. This effort was justified by the insights gained from making the corresponding shortcomings evident and people aware of them.

An example of the latter kind is the effort some people had to spend on syntactic data transformation issues and other issues around the invocation of actual SOAP based web services. Many participants were exclusively interested in solving problems on the semantic level and not interested in dealing with syntactic data transformation and network protocol issues. Some participants solved this conflict by teaming up with other groups whose interests were more in this area. But most participants felt they had to spend effort working on problems they were not really interested in just in order to be eligible to work on the problems they were actually interested in. This issue has influenced the design of recent scenarios as will be discussed further in Section 8.6.4.

Motivating Participation

Motivating participation turned out to be the most crucial factor in organizing a community based benchmarking initiative. Substantial effort for advertising the benchmarking campaigns were undertaken resulting in good levels of participation, even though still more participation would have been desirable. As usual, potential participants weigh the expected cost against the expected benefit. Both will be discussed in turn.

Costs: The cost regards primarily the effort involved in participation. This has been discussed above already. Apart from the effort, there was also monetary cost involved in participating in the Functional Scope Benchmark since the benchmarking methodology requires personal attendance at an evaluation workshop. This is different from the SWS matchmaking evaluation where the evaluation was performed offline and only results were presented at a workshop. Attendance at that workshop was encouraged to foster exchange and discussion of ideas but attendance was not mandatory.

The issue of workshop timing and location has been extensively discussed several times within the SWS Challenge initiative. In order to increase its visibility, the SWS Challenge workshops were aligned with major conferences in the field, in particular repeatedly with the European and International Semantic Web Conferences. But to reach out to different communities, workshops have also been held at the International Conference on Enterprise Information Systems of the software engineering community, the International Conference on Web Intelligence and Intelligent Agent Technology of the agents community and the European Conference on Web Services of the web services community. While this co-location involved a great deal of organization effort and, sometimes, monetary cost, it was not perceived to ultimately increase the level of participation significantly.

Some participants have argued in favor of the higher reputation of official conference workshops. However, others have preferred workshops that are only co-located with a conference without being part of the official conference program because of the significant registration fees involved with workshops being part of the official conference program. There was no clear picture with respect to higher participation in either one of both workshop types.

On a general level it appears that, compared to the S3 Contest, the necessity to travel to a workshop did prevent some participation in the SWS Challenge. On the other hand the repeated attendance at workshops within the SWS Challenge paid off by an increased coherence in the community and more intense discussion about the benchmark setup and results. These two effects, reduced but more intense participation, should be both considered when designing community benchmarks.

Apart from the monetary cost and the effort involved in participation the risk of poor benchmark results is probably a third factor that may cause reluctance to participate in the benchmarking. By definition, community benchmarking results are public. Comparatively poor results may be perceived as endangering the reputation of a group and ultimately even critical issues like the ability to raise further project funding. This is an inevitable dilemma which may be alleviated by presenting evaluation results in a proper form and communicating intensively with the stakeholders before evaluation results are published.

Benefits: The cost in terms of invested effort and funding as well as the risk of poor benchmarking results is counterbalanced by the perceived benefits of participating in the benchmarking. These include the gain of new scientific insights, increased visibility and credibility of research results and opportunities for publishing about both.

The primary goal of community benchmarking is to gain new scientific insights that ultimately lead to improved research results. All participants in the benchmarking reported that they found the participation to be an interesting and rewarding exercise through which they learned a lot. With this respect, the approach was clearly successful. Nevertheless, research is also directed by realistic, practical considerations around tangible benefits.

With this respect, proper publishing opportunities were repeatedly mentioned as an issue influencing participation levels. The first SWS Challenge workshops did not have official proceedings. This was perceived to be problematic by some participants. Consequently, later workshops aimed at having official proceedings (e.g., by IEEE). However, this did not seem to have a significant effect on participation. Furthermore, because of the cost involved, aiming at official proceedings was not even preferred by all participants. Like the first SWS Challenge workshops, the S3 Contest did not publish papers about the evaluation or evaluation results (even though participants were encouraged to submit such papers independently to other venues). This was mentioned by participants as an issue that should be improved.

Generally, the different nature of evaluation papers, which describe implemented solutions to common problems rather than novel technologies, has made it sometimes difficult to find proper publication venues. One of the consequences of the focused scope of evaluation workshops is typically a small number of submissions combined with a high acceptance rate. After all, the entrance barrier to such workshops consists of preparing a running correct solution to the given benchmark problems and not in presenting a sufficiently sound and novel theoretic paper.

However, a small number of submissions combined with a high acceptance rate is often perceived as an indication of low paper quality without sufficiently considering the non-standard nature of the workshop and the papers. Additionally, the

focus on implementation rather than theory was sometimes perceived as inferior to traditional papers, too. Therefore, the organization of journal special issues or similar high quality publication opportunities turned out to be rather difficult. The publication of a Springer book with consolidated results from the first year of the SWS Challenge ([PLZM08]) proved to be a good solution and was perceived very positively. A new edition of such a book including results from all evaluation initiatives (SWS Challenge, S3 Contest and WS Challenge) is in planning (as of 2009) and very positively perceived by the participants.

The most important tangible benefits of participation are an increased visibility and credibility of research results. Of course, these benefits grow with the acceptance and establishment of benchmarking as such and the specific benchmark in particular in the scientific community, i.e., they become most effective once a certain momentum is gained. The attractiveness of community benchmarking is biggest, once repeatable evaluations and credible validation of research results become almost mandatory in a community.

However, the survey of project based evaluations presented in Section 3.1.4 suggests that repeatable experimental evaluations are not yet mandatory in the area. Furthermore, experience from organizing the community benchmarking initiatives indicate that the esteem of experimental work in the area is still mixed. On the other hand, a trend towards higher esteem of such work can be observed in the area of SWS research as discussed in Section 1.2. Overall, prerequisites for successful community benchmarking seem acceptable with this respect, even though not optimal. It is also worth noting that similar problems have been reported from other areas of computer science, too. In the words of Feitelson:

> "In the context of reproducibility it may also be appropriate to challenge the prevailing emphasis on novelty and innovation in computer science, and especially in the systems area. Many leading conferences and journals cite originality as a major factor in accepting works for publication, leading to a culture where each researcher is motivated to create his own world that is distinct from (and incomparable with) those of others." [Fei06]

Feitelson also gives a number of examples for workshops and conferences devoted to empirical studies and states:

> "While this listing is encouraging, it is also disheartening that most of these venues are vary narrow in scope. Furthermore, their existence actually accentuates the low esteem by which experimental work is regarded in computer science. For example, the Internet Measurement conference web site states *'IMC was begun as a workshop in 2001*

> *in response to the difficulty at that time finding appropriate publication/presentation venues for high-quality Internet measurement research in general, and frustration with the annual ACM SIGCOMM conference's treatment of measurement submissions in particular'.*" [Fei06]

In the same way, the call for paper of the Annual Workshop on Duplicating, Deconstructing, and Debunking, held in conjunction with the International Symposium on Computer Architecture (ISCA) states:

> "Traditionally, computer architecture conferences and workshops focus almost exclusively on novelty and performance, neglecting an abundance of interesting work that lacks one or both of these attributes. A significant part of research — in fact, the backbone of the scientific method — involves independent validation of existing work and the exploration of strange ideas that never pan out."[1]

Without any doubt, a generally higher esteem of experimental, concrete work in computer science is the single most important key to encouraging people spending effort on evaluation and increasing participation in community benchmarking. As long as such experimental work is not sufficiently valued, comparative and experimental evaluations of existing technologies primarily involve more effort and less tangible benefits than work on novel technologies, even though they may finally have a stronger impact on the scientific progress of the field.

8.4. Discussion of Conceptual Framework

After having discussed the general community based approach to benchmarking and evaluation that this thesis followed, the concrete thesis contributions will be validated, starting with the validation of the conceptual framework for SWS technology evaluation presented in Chapter 4.

The engineering objectives for this framework were *comprehensiveness* and *well-foundedness*. Well-foundedness will be validated by discussing the methodological approach that was followed to derive the framework. Comprehensiveness will be validated by discussing the completeness of the framework and its practical applicability to relate different SWS evaluations to each other.

Additional validation of the conceptual framework is given through its publication by a high quality journal in the area (International Journal on Semantic Web and Information Systems [KKRPK08]). Additionally, this journal publication has been invited for an extended book chapter in the Advances in Semantic Web and Information Systems Book Series [KKRK10].

[1] `http://www.ece.wisc.edu/~wddd/`

8.4.1. Methodological Approach

The conceptual framework comprises two parts, the SWS evaluation criteria model and the SWS evaluation requirements catalogue, which have been derived following different methodologies. Both are discussed in turn.

SWS Evaluation Criteria Model

The SWS evaluation criteria model has been derived using the GQM approach from software engineering. This approach is an established practice for deriving evaluation criteria from engineering goals and has been in use for more than 25 years. Its validity will thus be assumed and not verified in this thesis. Rather we discuss its applicability to the problem and whether it has been applied in a sound way.

The GQM paradigm is a mechanism for defining and evaluating a set of operational goals, using measurement. It is is based on the assumption that the evaluation of any system should be an evaluation of fitness for purpose [Bas92, BCR94].

SWS are a family of technologies proposed to solve certain engineering problems stated in the introduction of this thesis. SWS evaluation in the context of this thesis is performed with the primary purpose of gathering knowledge about and supporting the iterative improvement of the evaluated tools with respect to the engineering problems SWS are supposed to solve. Thus, SWS evaluation should measure the advancement of the technologies with respect to these goals. The assumption of the GQM approach that an evaluation should be an evaluation of fitness for purpose is thus fulfilled and GQM can be properly applied to the problem of evaluating the achievement of the goals.

The engineering goals of SWS have been derived from a literature review of motivating use cases for SWS technology. This review was not exhaustive but still extensive. It is thus safe to argue that the goals have been properly identified. The goals were then operationalized by defining a set of questions whose answers characterize the fulfillment of each goal. Again, completeness of these questions can not be formally proven but the following section will argue in favor of the comprehensiveness of the framework.

The questions were analyzed with respect to their mutual relationships and five primary evaluation criteria were derived. While there is a certain freedom in the definition of evaluation criteria (for instance, one could treat performance and scalability either as separate or as one joint criteria) the derivation of evaluation criteria has been performed in a systematical and reasonable way.

SWS Evaluation Requirements Catalogue

The SWS evaluation requirements catalogue is based on the evaluation standards by the German Evaluation Society. These standards have been explicitly designed for the purpose of promoting evaluation quality in all application areas and are thus also applicable to the domain of interest. The standards have been passed by a broad committee in 2001 and are based on the standards by the US Joint Committee on Standards for Educational Evaluation and the standards of the Swiss Evaluation Society. They therefore form an established and high-quality framework of evaluation quality requirements based upon 20 years of evaluation experience [Bey03].

The requirements have been operationalized and adapted to the SWS technology evaluation use case via questions in the spirit of the GQM approach. These questions have been compiled based upon similar requirements and question catalogues from related work which, however, was not based on comparable standards. Basing the SWS evaluation requirements catalogue on a set of established standards and additionally integrating the viewpoints of several authors from related work ensures a maximum of objectivity and fairness of the resulting catalogue.

8.4.2. Completeness and Applicability of the Framework

Completeness of the SWS evaluation framework with respect to evaluation criteria and requirements can not be formally proven. However, a certain level of confidence can be established argumentatively.

Completeness of the evaluation criteria follows from the completeness of the identified engineering goals and the proper usage of the GQM approach to derive the previous from the latter. It is also indicated by the fact that existing evaluation initiatives could be properly represented within the dimension model (cf. Section 4.3). That said, the identified dimensions are expected to be complete, but specified on a relatively abstract level. This was done on purpose to ensure the wide applicability of the evaluation framework to all kinds of SWS evaluation.

Completeness of the requirements catalogue follows from the completeness of the evaluation standards the requirements are based upon. Confidence in the completeness of the requirements is further established by the integration of several similar requirements catalogues from related work. These were integrated without the need to extend the catalogue.

The applicability of the framework is demonstrated by using it to analyze the state of the art (Sections 4.3 and 4.4) and by using it as a framework to meta-evaluate the benchmarks contributed by this thesis (Sections 8.6 and 8.7). It will thus not be discussed here any further.

8.5. Discussion of Test Collection Development

Chapter 5 discussed requirements on SWS test collections and analyzed the available data. It was shown that existing data is far from meeting the desirable quality standards and argued that existing data needs to be shared more efficiently and new data should be developed collaboratively. Furthermore, it was discussed that one prerequisite for sharing data and obtaining contributions from the community is to offer appropriate tools that make contributing as easy and effortless as possible. This motivated the development and prototypical implementation of a portal (OPOSSum) that provides this tool support. The primary design goals of the portal were the following:

Goal 1: Promote exchange, reuse and collaborative improvement of existing data.

Goal 2: Improve structure, documentation, and usability.

Goal 3: Support reuse and comparisons across formalisms.

In the following, we briefly discuss whether these goals have been achieved. OPOSSum has been presented at the European Semantic Web Conference 2008 and the International Conference on Semantic Computing 2008, where it received the Best Demonstration Award [KKRK08a, KKRK08b]. As of January 2010, OPOSSum had 44 registered users and listed 2851 descriptions for 1524 services. This makes it the by far largest collection of SWS test data available and also highlights its successful adoption by the community. It can thus be confidently argued that OPOSSum successfully enables the exchange, reuse and collaborative improvement of existing data (Goal 1).

Nevertheless most of the data listed in OPOSSum has been added by the candidate. Adding of data does not provide an immediate benefit to the person who performs this task. Like all comparable community projects, the mutual benefit results from the sum of many altruistic contributions. This process usually takes a lot of time to gain a critical mass. Furthermore, adding data involves some overhead that not everyone appreciates. Partially, this is due to the fact that OPOSSum needs to be considered a prototypical implementation. The usability of the interface could be improved in a reimplementation that leverages modern Web technologies like JavaScript and Ajax.

Primarily however, this is also due to a conflict between the goal of making contributions easy (Goal 1) and the goal of making contributions effectively reusable (Goals 2 and 3). The more structure is enforced when new data is added to the portal, the more flexibly the data can be reused in different contexts. However, enforcing structure and metadata as opposed to allowing the simple dump of files or even archives also increases the necessary manual effort to add the data. With

respect to this tradeoff OPOSSum enforces a relatively high level of structure and metadata. Allowing for different such tradeoffs would probably increase the acceptance of the portal in the community.

However, with respect to structure, documentation, usability and reuse across formalisms the OPOSSum approach has proven very effective in the context of the SWS matchmaking evaluation (cf. Chapter 7). OPOSSum's highly structured relational model with lots of metadata allowed generating template descriptions for various formalisms when participants had to annotate the JGD with their formalism of choice. This relieved participants from a lot of tedious work and made the creation of alternative descriptions for the same set of services relatively easy (Goals 2 and 3). This capability surprised most participants and was very highly appreciated.

Also the development of the JGD, the assembly and management of the services with all the metadata, but in particular the collection and management of the reference judgments would have been entirely infeasible without OPOSSum. This collection was developed collaboratively and remotely by four people leveraging the capabilities of OPOSSum and serves as a demonstration of the usability and effectiveness of the OPOSSum approach. The JGD itself has been evaluated in Section 5.4 and will thus not be discussed here.

8.6. Meta-Evaluation of the Functional Scope Benchmark

Chapter 6 presented a benchmark for evaluating the functional scope of SWS discovery frameworks. This section meta-evaluates this benchmarks by four means.

First, it will be discussed whether the benchmark meets the design objectives that motivated its creation. Second, the benchmark will be assessed against the requirements catalogue for evaluations presented in Section 4.2. Third, the dissemination activities performed to promote the benchmark will be discussed. Fourth, the strengths and weaknesses of the benchmark will be discussed on a more general level with a specific focus on experiences and lessons learned during its execution within the SWS Challenge.

8.6.1. Achievement of Benchmark Design Objectives

The benchmark was designed to assess the functional scope and capabilities of SWS discovery frameworks (cf. Section 6.2). This concerns the tasks or phases during service discovery which are supported (for instance, whether information can be dynamically obtained from service endpoints and leveraged during the discovery or whether services can be fully automatically selected and invoked), the general capabilities of the discovery algorithms (for instance, whether the algorithms are

able to compose services if no single service is able to achieve the desired goal) and the practical expressivity of the frameworks (for instance, whether they support arithmetics or can correctly represent and handle aspects of date and time).

The benchmark aimed at providing means for a certification that offers an independent verification that claimed technologies actually work. Furthermore, it intended to explore the trade-offs among existing approaches and to figure out which parts of problem space may not yet be covered.

The repeated execution of the benchmark (cf. Section 6.6) illustrates the practical applicability of the benchmark with respect to the above listed engineering goals. The agreed upon problem scenarios allow the certification of technologies with respect to the associated functional challenges. Furthermore, the identification of these challenges supports the exploration of the problem space as such. Finally, the trade-offs among existing approaches are identified by means of the in-depth solution comparisons jointly prepared by participants in the evaluation.

Thus, the benchmark methodology is considered to achieve its design objectives. In practice, of course, the benchmark is limited to the problem space covered by existing problem scenarios. As discussed in Section 6.5.9, the claim is not that this coverage is complete. Completeness can only be achieved over time and is beyond the scope of this thesis.

The benchmark was designed under the primary constraints of not limiting or presupposing the technologies that it is applicable to in any way. This has been achieved by basing it exclusively on natural language descriptions and standard formalisms like XML and WSDL. In fact, the set of participants represents a variety of approaches, thus supporting the claim that the benchmark is indeed open to all kinds of technologies. The pros and cons of this setup as well as the general quality of the benchmark will be discussed in the following.

8.6.2. Assessment with respect to Requirements

The validation of the benchmark with respect to the requirements catalogue presented in Section 4.2 serves as an assessment of the benchmark's quality. Note that this assessment is performed with respect to the extended information available online[2] and not only with respect to the summarized information available in Chapter 6. Table 8.5 shows an overview of the results. A checkmark "✓" denotes that the requirement is fulfilled. A checkmark in parenthesis "(✓)" denotes that the requirement is partially fulfilled and that the benchmark should be further improved with respect to this requirement. A minus sign "–" denotes that the requirement is primarily not fulfilled and improvement is required.

[2] `http://sws-challenge.org`

Utility 1 (Stakeholder Identification)	✓
Utility 2 (Clarification of the Purposes of the Evaluation)	✓
Utility 3 (Evaluator Credibility and Competence)	✓
Utility 4 (Information Scope and Selection)	(✓)
Utility 5 (Transparency of Values)	✓
Utility 6 (Report Comprehensiveness and Clarity)	(✓)
Utility 7 (Evaluation Timeliness)	✓
Utility 8 (Evaluation Utilization and Use)	✓
Feasibility 1 (Appropriate Procedures)	(✓)
Feasibility 2 (Diplomatic Conduct)	✓
Feasibility 3 (Evaluation Efficiency)	(✓)
Propriety 1 (Formal Agreement)	(✓)
Propriety 2 (Protection of Individual Rights)	✓
Propriety 3 (Complete and Fair Investigation)	(✓)
Propriety 4 (Unbiased Conduct and Reporting)	✓
Propriety 5 (Disclosure of Findings)	✓
Accuracy 1 (Description of the Evaluand)	✓
Accuracy 2 (Context Analysis)	✓
Accuracy 3 (Described Purposes and Procedures)	(✓)
Accuracy 4 (Disclosure of Information Sources)	(✓)
Accuracy 5 (Valid and Reliable Information)	✓
Accuracy 6 (Systematic Data Review)	(✓)
Accuracy 7 (Analysis of Qualitative and Quantitative Information)	✓
Accuracy 8 (Justified Conclusions)	✓
Accuracy 9 (Meta-Evaluation)	(✓)

Table 8.5.: Assessment of the Functional Scope Benchmark

The table illustrates that all requirements are at least partially met by the benchmark. However, improvement with respect to ten of the twenty-five requirements is desirable. This also indicates that the requirements represent strict quality standards that can not be easily met. For improved readability, a discussion of all requirements is omitted here, but available in Appendix B.2. Here, only the limitations of the benchmark related to the ten requirements where further improvement is desirable will be discussed.

Problem Scenario Selection and Availability

Dynamic semantic service discovery is a visionary technology. Thus, the selection of problem scenarios can only to some extent represent problems found in actual

practice. Furthermore, while the selection of problem scenarios is approved through community consensus, it is supported by empirical work only in a very limited way (Utility 4). Coverage of the evaluation with respect to a complete investigation is limited by the availability of approved problem scenarios (Propriety 3). This is due to the significant investment in terms of time and effort required for the development of new problem scenarios. With this respect, there is also a tradeoff between enforcing high quality standards on new problem scenarios and having more such scenarios by making their addition easier.

Procedure Documentation

Documentation of evaluation procedures and problem scenarios is comparatively good, but could still be improved. While a lot of information about the benchmark is available, the corresponding information has been developed over years and requires restructuring and rewriting from time to time. Scattering of information limits the interpretability of evaluation results by outsiders (Utility 6). The older scenarios partially lack clear and rigid specifications of success criteria for certification. Some of the corresponding information is only implicitly available (Accuracy 3, Accuracy 9). However, the newer scenarios improve with this aspect already.

The benchmark description contains some information on the people responsible for the various tasks. However, no written formal agreement with this respect is available (Propriety 1). Attempts to formalize the implicitly acknowledged responsibilities have been undertaken but so far not succeeded.

The benchmarking methodology implies that significant parts of the actual evaluation are performed through live demonstrations, peer code reviews and corresponding discussions at workshops. So far, the workshops have not been sufficiently documented by minutes or other forms of recording. This limits the transparency of the information that the evaluation results are based upon (Accuracy 4) and makes meaningful meta-evaluations very difficult for outsiders (Accuracy 9).

Supporting Software

Two of the problem scenarios are accompanied by a testbed of actually working Web services. While these have been tested, more testing and improved documentation of the testing is clearly desirable (Accuracy 6). Furthermore, a higher degree of automation within the verification of the correctness of a problem solution would be helpful to make evaluations more efficient and more easily reproducible (Feasibility 1). Automation is currently still limited because of the resources required to implement corresponding tool support.

Involved Cost

No explicit discussion of the costs involved in participating in the benchmark is provided as part of the benchmark description. This involves monetary costs for traveling and workshop or conference registration fees as well as an estimate of the minimal effort required to prepare a solution to the problem scenarios (Feasibility 3).

8.6.3. Dissemination Activities

A brief discussion of the dissemination activities performed with respect to the benchmark serves as an assessment of whether the benchmark setup promotes a culture of collaboration and established structures to foster the co-evolution of the benchmarking efforts and the scientific community.

The benchmark has primarily been organized and developed within the greater scope of the SWS Challenge initiative. Within this context, eight workshops have been held between 2006 and 2009. Furthermore, there has been a special conference session about the comparative evaluation of SWS frameworks. A full listing and information is available online[3]. These events served for building a community actively interested in the matter that supports the successful co-evolution of the benchmark and the scientific community. Within the scope of the special session, papers comparing different solutions were prepared. These were jointly authored by the developers of the compared solutions, thus further promoting improved understanding of each others technologies and promoting a culture of collaboration. The comparison papers were extended and finalized for a book jointly written by the groups participating in the benchmark [PLZM08].

Furthermore, within the scope of the SWS Challenge a W3C Incubator Group has worked on the topic of developing a standard methodology for evaluating semantic web services based upon a standard set of problems and developing a public repository of such problems[4]. This group has published an Incubator Group report that also discussed the benchmark setup and methodology [PKMS08]. The Incubator Group gathered experts from many institutions, promoted the topic in the community and served as another important means of collaboration.

The benchmark has also been disseminated via the Semantic Institute International (STI[2]) Testbed and Challenges service which coordinates efforts around test beds and challenges for evaluating, testing, demonstrating, and comparing semantic technologies[5].

Finally, the benchmark has been presented as part of a tutorial on the evaluation of semantic web technologies at the 6th European Semantic Web Conference (ESWC

[3] `http://sws-challenge.org/wiki/index.php/Workshops`
[4] `http://www.w3.org/2005/Incubator/swsc/`
[5] `http://testbeds-challenges.sti2.org/`

2009)[6]. A second edition of this tutorial will be presented at the 7th European Semantic Web Conference (ESWC 2010)[7].

8.6.4. Strengths, Weaknesses and Lessons Learned

The meta-evaluation of the benchmark concludes with a critical discussion of its strengths and weaknesses and the lessons learned during its execution. The discussion will be centered on the evaluation measures and procedures. It covers measures and procedures on a general level, issues around making solutions to the benchmark problems publicly available or not, measures for the flexibility of solutions that did not work well and are thus not pursued anymore, the process of problem scenario development, the general design rationale of the problem scenarios and the way they are specified using natural language instead of some formal notation.

General Evaluation Measures and Procedures

The overall evaluation approach of the SWS Challenge initiative — certifying the ability of technologies based on solutions to common problem scenarios — has not been changed since its start in 2006. The presented benchmark represents the latest development of the discovery track within the Challenge. It differs in some important aspects from the setup originally envisioned since concrete measures and procedures being used have evolved over time. Here, we report on what has worked well, discuss the most common critique to the evaluation procedures still being used and outline why some measures originally being used have been abandoned meanwhile.

Generally, the approach of having a code review and corresponding discussions during the workshops has worked well. Originally the workshop has split into teams to verify the various solutions, but it turned out that everyone was interested in looking into everyone's solution. Thus, time permitting, solutions are currently evaluated by the workshop as a whole. We suspect that since evaluations are developed by the collective consensus of the whole workshop, they are better than they would have been had they been reached by smaller groups. Besides, we have found that expertise in understanding different technologies varies among the workshop participants and different people can examine different technologies more critically than others. Furthermore, all participants reported that the discussions about the solutions at the workshop greatly increased the mutual understanding for each others technologies and promoted the collaboration among teams.

[6] http://fusion.cs.uni-jena.de/professur/research/activities/sw-eval-tutorial-eswc09

[7] http://fusion.cs.uni-jena.de/professur/research/activities/sw-eval-tutorial-eswc10

However, there is also a flipside to the primarily manual verification of the solutions. First, the approach does not scale well to large number of submissions although this has not been a problem so far. More importantly, as has been mentioned above, the manual verification at the workshops is comparatively difficult to record in detail. This reduces the transparency and reproducibility of the evaluation results. This problem is aggravated since manual verification generally introduces some subjectivity and thus dependence from the actual team performing the verification. On the other hand, the implementation of a testbed that largely automates the correctness testing of a solution is extremely labor intensive and not even feasible in all cases. This is a continuous problem without an easy solution.

Especially in the beginning of the initiative, there was also a lack of formal requirements on the evaluation, leading to a great deal of discussion and some additional subjectivity in the evaluation results. The controversy usually manifested in numerous footnotes that had to be added to the evaluation results to indicate border cases of the evaluation. Apart from the subjectivity issue, these footnotes also made the evaluation results more difficult to interpret. To improve with this respect, more recent scenarios provide much more detailed specifications about the associated evaluation procedure and the necessary success criteria to pass a certification than the first ones did. Furthermore, some of the measures particularly vulnerable to subjectivity are not used anymore (this will be discussed further later in this section). This has improved the transparency and reproducibility of evaluation results significantly.

Participants were encouraged to upload their solution code and document it such that other people can reuse the code and independently rerun the solution. Apart from allowing everyone to learn about other approaches, this was also meant to improve the repeatability and verifiability of evaluation results. However, this process has not been very successful so far since most teams did not upload code or failed to document it to a level that made it useable by other people. This will be discussed in the following paragraph.

Code Upload and Solution Documentation

As mentioned, uploads of the implemented solutions to a public FTP server and the documentation of the solutions have been highly encouraged to improve transparency and reproducibility of evaluation results and allow everyone to learn about the certified technologies. However, this is an aspect that unfortunately has not worked particularly well so far. By far not all solutions have been uploaded and the documentation of the uploaded solutions is often insufficient to make them usable by other people. This does not only effectively prevent any reuse of existing solutions, it also prevents repeatability and renders an independent third party verification of the certification results largely infeasible.

Therefore, making the upload of the solutions mandatory and ensuring sufficient documentation has been discussed several times, but so far always been rejected, primarily for two reasons. First, requiring uploads of code is generally a controversial and difficult issue when it comes to industry participation and tools that are not necessarily open source. Code upload may be impossible for some participants due to legal and licensing restrictions. Second, enforcing code uploads and documentations further increases the barrier for participation which, given the large amount of effort involved in preparing a running solution, is already substantial. After all, primarily the concerns about an imbalanced relationship between costs and benefits associated with participation in the initiative tipped the scales towards not making the upload and documentation of solutions a mandatory requirement.

An alternative approach to mandatory public code uploads is the application of a repeatability check similar, for instance, to the SIGMOD repeatability guidelines introduced in 2008 [SIG07]. The basic idea is to have PC members independently verifying the usability and correctness of a solution based upon a confidential upload of code with sufficient accompanying documentation. However, this approach has not been implemented for practical reasons. Performing a remote verification on the machines of participants was considered to add little value over the demonstration anyway performed at the workshops. Forcing participants to provide their system on a certain platform (e.g., MS Windows) was perceived as a too severe barrier for participation. Offering verification on several common platforms by several PC members failed because too few PC members volunteered for providing this verification service. Consequently, the idea was not implemented.

Generally, code submissions and documentations, while a great idea in theory, so far proved infeasible for simple practical reasons. The whole issue is also a typical instance of the quality of evaluation being in conflict with the effort required for participation (or organization). It also illustrates a general deadlock regarding participation and commitment levels that is difficult to break. Without quality, new people (participants and organizers) are not attracted. But raising the entry barrier to increase quality may further discourage from commitment, leading to less participation, less critical mass and less attractiveness to new people.

Measuring the Flexibility of Technologies

Apart from certifying the functional scope and capabilities of technologies, the SWS Challenge originally also intended to measure the flexibility and adaptability of the participating technologies. The assumption was that extended use of formal semantics would lead to programs that are more adaptable and thus require less maintenance effort if the underlying problem scenario is changed. This assumption also concerns the expected competitive advantage of semantic technologies over conventional software engineering techniques [Pet06]. However, it turned out that

the objective testing of this assumption was largely infeasible within the context of an initiative like the SWS Challenge. Corresponding measures were thus not included in the benchmark presented in this thesis.

The first approach for measuring the adaptability of solutions was to evaluate the difficulty of moving from one problem level or sublevel to another. This was measured by determining whether code was changed or whether there was only a change to the declaration of objects upon which the code acted. Furthermore, it was tried to distinguish between whether the current declarations had to be altered or whether new declarations were simply added.

First of all, this approach required a code freeze of the original solution prior to the release of new problem levels. Otherwise, with knowledge about the nature of the upcoming changes, a solution can be designed in a way that makes implementation of changes easy. After all, the very nature of change is that it has not been foreseen. However, the necessary code freeze was difficult to implement, in particular since all participating systems were under active developments during the participation. Changes resulting from the regular ongoing development of the technologies and changes resulting from the changes of the problem scenarios were difficult to distinguish. On the other hand, the maintenance of different development branches imposed an additional burden on the participants that most were not willing to take.

Apart from the difficulties involved with the necessary code freezes, it was found that the distinctions between changes in code and data could not be made objectively. For interpreted languages like Lisp, for instance, there simply is no objective difference between declarations and code. The differentiation also proved extremely problematic in those approaches, where programs are specified graphically in a workflow like fashion and then automatically compiled to executable code. Again, there was no objective basis to differentiate between declarations and imperative code. Similarly, one approach formalized the problem semantics in Java which allowed embedding Java code fragments into ontological instances in an object oriented manner. Once more, it turned out to be impossible to objectively decide which was code and which was data. It was tried making a collective consensus on simply whether code or declarations have been changed but, not entirely surprising, this resulted in extensive discussions until it was finally agreed that there is simply no basis to objectively judge the difference between changes in the data and the code and the corresponding measure was dropped.

The second approach, named “surprise problem”, was primarily motivated by the desire of avoiding the code freezes. Short before a workshop, an altered version of a scenario was released with limited time left to implement it prior to the workshop. It was hoped that the limited time frame would render the creation of a new solution from scratch impossible, thus forcing participants to implement changes on top of the existing original solutions. However, at the Stanford 2007 workshop, one

participant was able to program the surprise problem from scratch in Java and present a verifiable solution in about two hours. This highlights a fundamental problem related to testing the adaptability of programs.

Small problem scenarios of limited size and complexity can be quickly and efficiently programmed from scratch by a skilled programmer, in fact, typically more quickly and efficiently than using some sophisticated frameworks making use of formal semantics and reasoning. The overhead involved in more formal methods has a chance to pay off only in very complex and large scenarios that a single programmer can no longer easily comprehend. However, the usage of such scenarios for evaluation purposes is entirely infeasible, especially in academic settings. Spending several person months of full time labor to work on a complex evaluation scenario is usually simply no option. Therefore, evaluation scenarios have to be of limited size and complexity. In such settings, formal methods can not compete with ad-hoc programming in terms of programmer productivity.

Notably, this problem persists if time and lines-of-code-changed, the two obvious measures for programming effort are employed. Additionally, counting the lines of code being changed is again very difficult in cases where programming is not performed by typing instructions into a text editor, but by graphically changing properties or the flow of a workflow, for instance. Directly measuring the time necessary to implement changes is generally difficult to control unless changes are made live during a workshop. However, enforcing the implementation of changes live during a workshop or conference proved highly unpopular with participants. Furthermore, there were worries that measuring the time would lead to an unwanted competitive atmosphere during the evaluation in the first place. In summary, it does not seem as if there is yet a good, reliable and efficient measure for the adaptability and flexibility of solutions which can be feasibly evaluated within the limits of a community evaluation initiative.

Problem Scenario Development

The actual evaluation measure being employed by the benchmark is to assess the ability to solve functional challenges, each represented by one or more given concrete problem scenario levels. Therefore, the set of problem scenarios forms the concrete measuring instrument of the benchmark.

With this respect, we learned that, even apart from the actual testbed implementation, the development of scenarios on the theoretical level proved to be significantly more complex than estimated. Finalizing an initial problem idea into a realistic, well specified problem scenario, is a long process. The necessary balancing between a trivial problem that lacks relevance for real world settings and a full-blown realistic problem that is infeasible to solve in academic settings has been a continuous challenge with all scenarios.

The formal process of scenario contributions which involves presentation and discussion at a workshop and among the SWS Challenge organizing team required further effort on side of scenario contributors. While these discussions greatly stimulated scientific debate about the matter, led to an increase of understanding by all participants and generally paid off by more well defined scenarios, they also increased the development time until a scenario could be finalized. In fact, the Logistics Management Scenario was first presented in June 2008 [CCC+08], but ready for evaluation only more than one year later at the SWS Challenge workshop in Eindhoven, The Netherlands, November 2009. Yet, despite of several iterations during the development of all scenarios, it turned out that none of the scenario specifications was entirely free of bugs when released.

Maintaining a proper quality management within the constraints of largely unfunded work and without discouraging new scenario proposals presents a continuous challenge, in particular since ambiguous and controversial scenarios or error-prone testbeds harm the reputation of the benchmark and may discourage potential participants. This, however, is a general problem of all similar benchmarking initiatives as long as the development of such benchmarks is not explicitly funded.

Design Rationale of Problem Scenarios

The original approach to scenario design was to create a hierarchy of problem levels where each problem builds upon the previous one and adds a piece of complexity. The working hypothesis was that "we should build up a giant macro scenario out of our individual scenarios. This is intended to be a complex multiple customer/manufacturer/multiple supplier/multiple shipper problem with complex product configuration constraints and goals. The hypothesis is that a problem change with such a complex scenario will differentiate software technologies and reveal advantages of a subset in modifying such a complex application" [PKM+08]. This did not prove to be a particularly feasible approach. Therefore, the design of problem scenarios has changed over time, motivated by reducing the effort involved in participation and by making the evaluation results more transparent and more easily to interpret.

As mentioned previously, different approaches were challenged by entirely different problem aspects. The inclusion of dynamic information into the matchmaking process, for instance, had been considered during the design of the DIANE framework leveraged by the Jena solution from the very beginning. Therefore, this functional challenge did not pose significant difficulties to this framework, despite of being initially challenging for most others. In contrast, the evaluation of even simple rules to compute the price of a shipment based upon the destination address and the weight of the parcel was not supported by the DIANE framework directly, but did not pose a challenge to other approaches [KKRK06b, BCC+06b]. Overall, it turned

out to be impossible to design a problem hierarchy such that the level of difficulty increases relatively equally for all approaches.

As a consequence, the current approach to scenario design is to separate concerns as far as possible. I.e., problem levels are designed such that, ideally, there is a largely independent problem level for every functional challenge which focuses on this challenge exclusively. Even though not always possible, this has two important advantages.

First, it allows a more fine grained evaluation and makes evaluation results more easily to interpret since there are less potential causes when an approach fails solving a particular problem level. The Jena solution, for instance, failed to solve a problem level designed to test for the capability to compose services in order to purchase a set of different but compatible products from several vendors. However, the failure was not due to a lack of composition capabilities, but an insufficient ability to process list-based attributes during the matchmaking (the information about the compatibility of products was specified using list-based attributes enumerating the products compatible with a particular other product) [KKR07c]. As is illustrated by this example, the mixing of different challenges (in this case, list based attributes and correlated composition) results in evaluation results that are not always intuitively interpretable anymore.

The second advantage of the altered scenario design rationale is that participants can more freely choose the problem aspects of interests to them. They are no more required to solve "basic" problems which may be out of their scope and may also be far from trivial for them, just in order to participate on the problem levels they are really interested in. This reduces the effort involved in participation and significantly lowers the entrance barrier to the evaluation. This may be illustrated by a concrete example. Originally, the SWS Challenge has applied a principle called "no participation without invocation". This referred to the fact that all scenarios were backed by running web service implementations and that all scenario solutions had to interact with these service implementations [PKM+08]. The principle was motivated by the desire of making the evaluation more realistic and relevant by forcing solutions to actually deal with existing standard technologies (XML, SOAP web services).

However, as a result, all participants had to spend major effort on grounding semantic descriptions to XML and linking semantic frameworks with service execution engines able to execute web service calls. These aspects were not necessarily in the focus of the participating groups, in fact, two groups teamed up with other groups in order to divide the labor and separate the concerns of the semantic layer from those of the execution layer [BCC+06a, KMK+08]. Overall, while also having its benefits, the principle created a significant barrier for entrance to the evaluation. Therefore, the principle was partially abandoned. While the Shipment Discovery Scenario and the Hardware Purchasing Discovery Scenario still require calling web

services, the Logistics Scenario is restricted to discovery based purely on static service descriptions. It does not require interacting with actual web services.

Natural Language Scenario Specifications

Another principle of the benchmark is to not formalize the problem scenarios using logic or formal notations, but to specify the semantics of the problem scenarios using natural language only (the interfaces of the associated testbed services are specified using WSDL and XML schema). It is believed that how to properly formalize a problem domain, i.e., the correct representation of the explicit and implicit domain knowledge, the necessary and sufficient detailedness of the formalization and the choice of the most suitable representation formalism still constitutes an important and largely open research problem. Therefore, the problem formalization should be object of the evaluation and not be dictated by the benchmark. Furthermore, making choices regarding the problem formalization as part of providing the testbed tends to impose a specific solution approach on the participants. This may entirely exclude certain approaches and generally contradicts the open approach of providing an unbiased, level playing field for all technologies.

Among the participants, however, this issue tended to be controversial. Some participants have repeatedly called for formal problem scenario specifications to avoid ambiguities and ease the development process of problem solutions. Contrary, other participants have argued that formal specifications may be more difficult to use and understand. Additionally, requiring a normative, formalized version of the problem scenarios also increases the effort for contributors of new scenarios.

It is also worth noting that participants were encouraged to share their non-normative formalizations of the problem scenarios for reuse by others. It was hoped that this way, some formalizations would be reused more often than others, leading to the identification of a best of breed formalization. However, so far, not much reuse of formalization has occurred. Many participants use non-standard formalisms that can not be easily reused by other participants. Furthermore, as mentioned above, the level of sharing and documentation of solution was generally low. However, it may also be that participants did not want to rely on non-normative formalizations by other participants and thus preferred creating their own formalizations.

With respect to the inevitable ambiguities involved in natural language scenario specifications we learned that so far, none of the scenario specifications was entirely unambiguous upon release. However, during the usage of a scenario, ambiguities are usually discovered and can be resolved subsequently. Thus, over time, scenario specifications, even natural language ones, become increasingly well-specified. Furthermore, we suspect that even formal scenario specifications will initially often be underspecified, thus facing similar problems like those involved with natural

language specifications. Overall, it is believed that the usage of natural language specifications has worked well.

8.7. Meta-Evaluation of the SWS Matchmaking Benchmark

Chapter 7 presented a benchmark for evaluating SWS matchmakers. This section meta-evaluates this benchmarks by four means.

First, it will be discussed whether the benchmark meets the design objectives that motivated its creation. Second, the benchmark will be assessed against the requirements catalogue for evaluations presented in Section 4.2. Third, the dissemination activities performed to promote the benchmark will be discussed. Fourth, the strengths and weaknesses of the benchmark will be discussed on a more general level.

Note that the reliability of the benchmark's retrieval correctness measures has already been extensively discussed in Section 7.9. That discussion covered the effect that different definitions of relevance, subjective relevance judgments and different retrieval correctness measures have on the evaluation results. These issues will not be covered here again.

Achievement of Benchmark Design Objectives

The benchmark was designed to investigate a set of questions stated in Section 7.2. First and foremost, the benchmark aimed at investigating how precise, complete and efficient current technologies for service retrieval are. The execution of the benchmark presented in Section 7.8 illustrated that the benchmark is suitable for answering this question. Threats to the benchmark result's validity and reliability have been extensively discussed in Section 7.9.

Based upon the primary objective of making the retrieval correctness of current service retrieval technology measurable in a reliable way, the benchmark additionally aimed at providing clues with respect to other important questions like the right level of detail to describe services, the best pattern or formalism to do so or the properties of certain services and queries that make correct retrieval difficult.

By making the retrieval correctness of entirely different matchmakers measurable, the benchmarking methodology succeeds in paving the way to answering such questions. However, as stated at the beginning of Chapter 7, concrete answers to these questions are beyond the scope of this thesis for two reasons. For one thing, reliable conclusions with respect to such questions require more data than became available through the initial execution of the benchmark performed within the scope of this thesis. Repeated execution of the benchmark with different test collections

resulting in much more data is required before such questions can be answered properly. For another thing, conclusions about complex and widely different technologies are tricky. They require in-depth analysis of the benchmark results based on intimate knowledge about an evaluated technology to track down observed performance characteristics to their causes in the test data and the evaluated technology. Such analysis is beyond the scope of this thesis (cf. Section 7.2). Rather, this thesis attempted to provide a methodology that can be applied to gather sufficient data to be able to perform such analysis in principle.

With these restrictions, the benchmark is considered to suitably answer the evaluation questions that motivated the benchmark design.

Besides the desire of answering the evaluation questions of interest, the benchmark was motivated by shortcomings in existing similar SWS matchmaker benchmarking approaches (cf. Section 7.3). Achievement with respect to improving these shortcomings will be discussed now.

First, previous work relied on specific description formalisms and did not allow a comparison of matchmakers across formalisms. The benchmarking setup successfully resolved this issue. This is illustrated by the variety of matchmaking approaches based upon entirely different semantic models that participated in the benchmark execution.

Second, previous work did not allow to make the formalism itself subject of the evaluation. Achievement with respect to this shortcoming directly follows from the previous one. The benchmark allows the comparison of matchmakers across formalisms. It thus allows making the formalism subject to the evaluation. However, careful and knowledgeable analysis of the evaluation results is necessary to separate performance issues resulting from the used formalism from those resulting from the specific algorithm operating on the descriptions and performing the matchmaking.

Third, evaluation results obtained using state of the art benchmarking approaches may be compromised, if there is a lack of alignment of the modeling approach represented by the provided service descriptions with what the various evaluated matchmakers expect. This problem was circumvented by letting the participants create their own annotations for a set of services specified in natural language. This guaranteed proper descriptions for each evaluated matchmaker to the greatest possible extent.

In summary, the design objectives of the benchmark are considered to be achieved.

8.7.1. Assessment with respect to Requirements

The validation of the benchmark with respect to the requirements catalogue presented in Section 4.2 serves as an assessment of the benchmark's quality. Note that this assessment is performed with respect to the extended information avail-

able online[8] and not only with respect to the summarized information available in Chapter 7. Table 8.6 shows an overview of the results. A checkmark "✓" denotes that the requirement is fulfilled. A checkmark in parenthesis "(✓)" denotes that the requirement is partially fulfilled and that the benchmark should be further improved with respect to this requirement. A minus sign "–" denotes that the requirement is primarily not fulfilled and improvement is required.

The table illustrates that all requirements are at least partially met by the benchmark. However, improvement with respect to five of the twenty-five requirements is desirable. For improved readability, a discussion of all requirements is omitted here, but available in Appendix C.1. Here, only the limitations of the benchmark with respect to the mentioned five requirements where will be discussed.

Test Data Availability

This benchmark, even more so than others, critically depends on the volume of test data available and being used. The test data offered by the benchmark is realistic and supported by empirical work. However, it has not been fully leveraged in the actual benchmark execution and more data covering a greater variety of data characteristics is desirable. More test data would allow a more complete and thus also fairer investigation of the performance of the benchmarked tools (Propriety 3).

Procedure Documentation

Procedures and the setup of the benchmark are generally specified in detail and properly documented. However, a formal agreement about the obligations of all parties involved in the execution of the benchmark has not been prepared (Propriety 1).

Supporting Software

Two distinct tools support the benchmark. All aspects around test collection development were handled within the OPOSSum portal. This portal is open source and freely accessible. The actual benchmark execution was supported by the SME2 evaluation environment. This eased the process of measurement and analysis of results considerably. SME2 is available free of charge but not open source. Thus, the correctness of the software and corresponding data analysis can not be easily verified (Utility 6). Furthermore, it makes extensions of the data analysis somewhat difficult (Utility 8).

While much of the data analysis was automated via SME2, not all measures used and analyses performed are yet supported via SME2. The corresponding functional-

[8] `http://fusion.cs.uni-jena.de/professur/jgdeval/web-service-retrieval-evaluation`

Utility 1 (Stakeholder Identification)	✓
Utility 2 (Clarification of the Purposes of the Evaluation)	✓
Utility 3 (Evaluator Credibility and Competence)	✓
Utility 4 (Information Scope and Selection)	✓
Utility 5 (Transparency of Values)	✓
Utility 6 (Report Comprehensiveness and Clarity)	✓
Utility 7 (Evaluation Timeliness)	✓
Utility 8 (Evaluation Utilization and Use)	(✓)
Feasibility 1 (Appropriate Procedures)	(✓)
Feasibility 2 (Diplomatic Conduct)	✓
Feasibility 3 (Evaluation Efficiency)	✓
Propriety 1 (Formal Agreement)	(✓)
Propriety 2 (Protection of Individual Rights)	✓
Propriety 3 (Complete and Fair Investigation)	(✓)
Propriety 4 (Unbiased Conduct and Reporting)	✓
Propriety 5 (Disclosure of Findings)	✓
Accuracy 1 (Description of the Evaluand)	✓
Accuracy 2 (Context Analysis)	✓
Accuracy 3 (Described Purposes and Procedures)	✓
Accuracy 4 (Disclosure of Information Sources)	✓
Accuracy 5 (Valid and Reliable Information)	✓
Accuracy 6 (Systematic Data Review)	(✓)
Accuracy 7 (Analysis of Qualitative and Quantitative Information)	✓
Accuracy 8 (Justified Conclusions)	✓
Accuracy 9 (Meta-Evaluation)	✓

Table 8.6.: Assessment of the SWS Matchmaking Benchmark

ity had to be implemented outside of SME2 and some of the analysis was performed manually. More extensive automated data analysis within SME2 (or a comparable environment) would be desirable, but has not been provided yet due to limited resources (Feasibility 1).

8.7.2. Dissemination Activities

A brief discussion of the dissemination activities performed with respect to the benchmark serves as an assessment of whether the benchmark meets the requirements of promoting a culture of collaboration and establishing structures to foster the co-evolution of the benchmarking efforts and the scientific community.

The concept of the benchmark was discussed extensively within the S3 Contest and the related SWS Challenge communities. It was publicized repeatedly via open calls for feedback and participation on public mailing lists, through direct personal contact via email and at conferences and via a tutorial on Semantic Web technology evaluation at the 6th European Semantic Web Conference.

Furthermore the benchmark execution was organized as a complementary track under the wider scope of the established S3 Contest initiative, which has been held yearly since 2007 in conjunction with the International Semantic Web Conference (ISWC). Evaluation results were presented at the corresponding workshop at ISWC2009. The S3 Contest is further disseminated via the Semantic Institute International (STI^2) Testbed and Challenges service which coordinates efforts around test beds and challenges for evaluating, testing, demonstrating, and comparing semantic technologies[9].

Feedback from participants of the benchmarking was collected repeatedly during the planning phase, the execution of the benchmark and after results were prepared. This led to extensive discussion and exchange of experiences and ideas among participants. A book on SWS evaluation with contributions by the participants of the evaluation is currently being organized.

Finally, the benchmark has been presented as part of a tutorial on the evaluation of semantic web technologies at the 6th European Semantic Web Conference (ESWC 2009)[10]. A second edition of this tutorial will be presented at the 7th European Semantic Web Conference (ESWC 2010)[11].

[9] http://testbeds-challenges.sti2.org/

[10] http://fusion.cs.uni-jena.de/professur/research/activities/sw-eval-tutorial-eswc09

[11] http://fusion.cs.uni-jena.de/professur/research/activities/sw-eval-tutorial-eswc10

8.7.3. Strengths, Weaknesses and Lessons Learned

The meta-evaluation of the benchmark concludes with a general discussion of the benchmark. Overall, the benchmarking has worked very well and was very positively commented by the participants. In fact, there was no critique to the general setup of the benchmark by the participants. We therefore focus on discussing a few choices and lessons learned that concern the availability and publication of test data, the presentation of the benchmark results and the creation of extended analyses based upon these results.

Availability and Publication of Test Data

As mentioned, the availability and quality of test data is particularly crucial for this benchmark. With this respect, it was unfortunate that the 200 services of the JGD overcharged participants and the benchmark had to be executed on only 50 services. It also highlights that reducing the effort necessary for annotating services is quite essential. The creation of service description templates for each participant from the structured data available via OPOSSum was thus very highly appreciated by the participants and proved very useful.

A difficult issue with respect to the test data is whether it should be made fully public or not. Publishing all test data allows everyone to reproduce and verify evaluation results and to use the data in other, possibly unforeseen contexts. On the other hand, the setup of the benchmark requires that services and requests are formalized independently by different people. This can no longer be enforced if the test data is public. The data would still be useful to people that want to learn about the strengths and weaknesses of their technologies, but the reliability of evaluation results would be reduced, if people with access to the service requests and relevance judgments started to optimize their descriptions or algorithms towards this data.

The compromise that was chosen for this benchmark is to give participants full access to all data such that they can verify their own evaluation results and run additional tests during the process of improving their technology. Other people, however, are given access to the service requests and relevance judgments only upon request. While this approach does not guarantee the privacy of that data, it is hoped that it will limit the circulation of the collection sufficiently such that it can still be used reliably in a second iteration of the benchmarking.

Presentation of Benchmark Results

It turned out that presentation of benchmark results was a more sensitive issue with this benchmark than with the Functional Scope Benchmark. It seemed that participants associated notions of inferiority or superiority to a greater extent than with the other benchmark. This resulted in some iterations in how the benchmark results

were presented until eventually all participants were satisfied. We believe that the approach of letting participants approve and if necessary veto the presentation of their results increased the satisfaction of people with the benchmark and thus the attractiveness of participating in a new edition of the benchmarking.

Another issue related to the presentation of benchmark results is whether attendance at the workshop at which the results are presented should be mandatory or not. Since attendance is not crucial for the actual evaluation, workshop attendance is optional within the S3 Contest initiative. This was perceived positively by participants, many of which otherwise might not have participated in the evaluation. Nevertheless this also resulted in few participants actually attending the workshop. While there still was valuable discussion about the evaluation results via email, it is believed that a face-to-face meeting would promote collaboration and further benchmark development.

Analysis of Benchmark Results

An important difference between the execution of this benchmark within the S3 Contest and the execution of the Functional Scope Benchmark within the SWS Challenge is whether papers describing the participating technologies are prepared or not. This is mandatory within the SWS Challenge and has previously not been encouraged within the S3 Contest.

However, the actual usefulness of evaluation results increases dramatically if participants make detailed analyses of the causes for the measured performance characteristics available. Otherwise, learning from each other's approaches is rather difficult. On the other hand, the preparation of papers is difficult to enforce since they can only be written after the actual evaluation took place if they are supposed to provide analyses of the evaluation results and not mere descriptions of the evaluated systems which are typically already available in existing publications.

However, a survey among the 2009 S3 Contest participants showed that all of them were interested in contributing chapters with analyses of their evaluation results and what they learned from them to a corresponding book project. It is believed that promoting such analyses by offering a suitable publication opportunity is very valuable for leveraging the evaluation results and promoting their use in a wider community.

8.8. Summary

This chapter presented the validation of the thesis. First, the overall thesis approach of organizing evaluations as community initiatives was validated by verifying whether the necessary prerequisites to successfully perform community based

benchmarking exist. Furthermore, experiences and lessons learned during the process of organizing community-wide benchmarking initiatives were discussed. Subsequently, the concrete thesis contributions presented in Part II were validated and the achievement of the thesis objectives was verified.

The conceptual framework for SWS technology evaluation was assessed with respect to its methodological sound derivation and its completeness and usefulness to relate and discuss SWS technology evaluation approaches. Following, the contributions towards tool support for better test collections were evaluated by discussing whether the tools provide the desired functionality. Finally, extensive meta-evaluations of the two benchmarks contributed by this thesis were performed. These comprised an assessment of whether the benchmarks achieve the engineering objectives that motivated their creation, a review of the benchmarks quality with respect to the requirements catalogue from the conceptual framework, a report of the dissemination activities that were performed to promote the usage and approval of the benchmarks by the community and a discussion of the benchmarks' strengths, weaknesses and general lessons learned.

The validation results support the claim that this thesis laid the foundation for a more systematic and thorough treatment of the subject and contributed well-founded means to reliably evaluate selected aspects of SWS technology. Yet, it could not and was not meant to provide a complete solution to SWS benchmarking or give definite answers to fundamental questions about the most suitable formalism or approach to SWS. Thomas A. Edison's remark "We really haven't got any great amount of data on the subject, and without data how can we reach any definite conclusions?" reminds of the necessity to gather more data to achieve greater confidence about any conclusions being drawn. The need for benchmarking to be a continuous effort in order to support and promote scientific progress was stressed throughout this thesis several times anyway. The conclusion in the next final chapter discusses the outcomes of the thesis in more depth and describes directions of future work.

CHAPTER 9

Conclusions and Outlook

> When you can measure what you are speaking about and express it in numbers, you know something about it; but when you cannot measure it, when you cannot express it in numbers, your knowledge is of a meager and unsatisfactory kind.
>
> *(Lord Kelvin)*

In this final chapter, we summarize the motivation, the contributions and the main results of the thesis. Subsequently, we describe directions for future work in the area and conclude.

9.1. Summary

The following summary recapitulates the motivation of the thesis, presents its main contributions and finally briefly discusses its validation.

9.1.1. Motivation

The thesis was motivated by the conviction that more thorough experimental evaluation of SWS technology is required for the future scientific development of the area and the adoption of the developed approaches by industry. This conviction follows examples from the history of related areas, like information retrieval, where the existence of agreed upon standard benchmarks resulted in much more rapid progress. Further motivation follows from a theory of benchmarking which predicts rapid technical progress and community building to be associated with the creation and widespread use of a benchmark within a research area. The theory further

argues that benchmarking can and should be actively pursued in order to benefit from these positive consequences instead of just enjoying them as side effects.

A survey of the state of the art showed that there are some promising community evaluation initiatives, but that they are in an initial state with several shortcomings which are not discussed sufficiently. Generally, research contributions in the area are often insufficiently validated, in particular experimentally. This may largely be due to a lack of established evaluation setups and benchmarks. The few existing evaluation efforts, in papers, projects as well as the mentioned community evaluation initiatives address only aspects of the whole process. They typically fail to put their evaluation approach into a wider context and to discuss and assess it critically. Overall, a thorough and comprehensive discussion of the nature of evaluation in the area of SWS, including the identifications of the criteria to evaluate and the definition of standards to promote evaluation quality was lacking.

Therefore, the objective of this thesis was to develop a comprehensive and well-founded conceptual model for SWS technology evaluation which identifies evaluation criteria as well as requirement standards to promote and ensure evaluation quality. Furthermore, reference benchmarks for selected evaluation criteria and use cases that solve concrete benchmarking needs in the area should be developed and executed.

9.1.2. Contributions

To lay the foundation for a well-founded treatment of SWS evaluation, the envisioned conceptual model for SWS technology evaluation was provided. The Goal-Question-Metric approach from software engineering was employed to derive the possible criteria dimensions from a goal analysis of SWS technology. The criteria model allows classifying evaluation approaches and putting them into context in a well-structured way. Furthermore, a detailed catalogue for requirements to SWS technology evaluation was defined. This requirements catalogue promotes evaluation quality and supports a thorough meta-evaluation of SWS technology evaluations. It has been derived from evaluation standards published by the German Society for Evaluation and adapted and operationalized using input from a literature review on evaluation requirements in related areas.

Having introduced the conceptual model for SWS technology evaluation, a more detailed structured additional analysis of the state of the art has been performed. This analysis discussed how existing approaches address evaluation along the previously derived evaluation criteria. It also included a detailed assessment of three community evaluation initiatives with respect to the evaluation requirements catalogue. The discussion and meta evaluation of current approaches resulted in the identification of open problems, existing shortcomings and possible improvements. From the open problems, three were selected for which the thesis contributed con-

crete solutions. First, means for obtaining meaningful test data for SWS technology evaluation were established. Second, a benchmark for assessing the functional scope of SWS discovery frameworks was developed. Third, another benchmark for evaluating SWS matchmakers was developed. Each of these contributions will be briefly described in turn.

Test Data for SWS Evaluation

Based upon a requirement analysis for SWS test data, the available data was surveyed. This survey covered SWS descriptions generally visible on the Web as well as those in collections explicitly created for evaluation purposes. It was found that there is too little existing data, in particular for some formalisms like WSMO/WSML and that the existing data suffers from several shortcomings regarding its quality and diversity. It was argued that community involvement is crucial for successfully building high-quality test collections because of the tremendous effort involved and in order to achieve diversity and impartiality of the created test data. Such community involvement is only possible if proper tool support is available.

Therefore, OPOSSum, a portal for distributing and collaboratively creating and improving SWS test data was developed. The portal's design objectives were promoting the exchange and collaborative improvement of existing data, improving structure, documentation and usability of the data as well as supporting reuse and comparison across formalisms. The portal was developed as a publicly available, open source web application and publicized at several events. The primary existing collections were integrated and thus made available in a more usable way than previously. Furthermore, the Jena Geography Dataset, a new test collection that improved over the state of the art with respect to the requirements on SWS test data was developed within the portal and contributed.

Benchmarking the Functional Scope of SWS Discovery Frameworks

Following the work on SWS test data, a benchmark for assessing and certifying the functional scope and capabilities of SWS discovery frameworks was presented. The benchmark defines a set of service discovery problem scenarios specified in natural language. Each scenario defines a set of concrete service offers and is structured into a set of problem levels. Each level is represented by a number of specific service requests. Furthermore, a list of fundamental functional challenges, like the capability to represent numbers, perform arithmetic computations, express and evaluate complex preferences or dynamically gather information from web service endpoints has been defined. The problem levels from the scenarios are associated with the functional challenges that need to be solved in order to correctly process the service requests from the corresponding level.

The benchmarking approach consists of letting people present their solutions to the problem scenarios at evaluation workshops. The workshop verifies correctly solved problem levels and thus certifies functional challenges that an approach is capable of correctly dealing with. Furthermore, the methodology comprises soliciting papers that discuss the differences between approaches by comparing concrete solutions to the common problem scenarios. These papers are jointly written by the authors of the compared solutions and promote a deeper understanding of each other's approaches and the involved tradeoffs. The benchmarking approach has been successfully implemented and executed within the SWS Challenge community evaluation initiative over several years. Corresponding results and comparisons have been presented.

Benchmarking SWS Matchmaking

The second benchmark contributed by this thesis deals with evaluating SWS matchmakers. It improves the state of the art in three key areas. On a practical level, it employs richer and more realistic test data. On a methodological level, it contributes a novel evaluation setup that allows comparing matchmakers across formalisms, models realistic environments where service offer and request descriptions are developed independently and supports an analysis of the effects of using different description formalisms and descriptions with differing detailedness. On the analytical level, it provides improved measures that support a more detailed, reliable and fine-grained retrieval correctness analysis.

To this end, issues around relevance for service retrieval were investigated. Different relevance models, in particular novel ones based on non-binary relevance were presented and an experiment on the reliability of relevance judgments obtained from human assessors was performed. The experiment showed high inconsistency in the commonly used human relevance judgments and a dependency between the used relevance model and the observed level of inconsistency. A methodology for obtaining reliable judgments was presented.

Furthermore, several retrieval correctness measures from information retrieval that leverage additional information contained in non-binary relevance judgments were introduced and discussed with respect to their reliability. A number of shortcomings were identified and a set of improved measures proposed.

The benchmark was implemented and executed under the umbrella of the closely related S3 Contest community initiative. Based upon the benchmarking results, an in-depth analysis of the reliability of the benchmark was performed. The analysis comprised the effects of different relevance models, inconsistent relevance judgments and different evaluation measures on the evaluation results. It was found that binary relevance is highly sensitive towards changes in the relevance model and should be used with caution. Graded relevance was found to be much more reliable and should

thus be preferred. Inconsistency in relevance judgments seems to affect evaluation results generally only very moderately. Finally, the chosen correctness measure was found to have significant influence on the evaluation results. The parallel usage of different measures is thus recommended. These findings are expected to make SWS matchmaking correctness evaluations much more reliable and meaningful in the future.

9.1.3. Validation

The contributions of the thesis were validated by various means. As a prerequisite, a critical appraisal of the emphasis on community involvement to SWS technology evaluation was performed. This included an assessment that SWS related research is ready for community based benchmarking and a comprehensive discussion of lessons learned during the organization of two community-wide evaluation initiatives. The community focused approach was found to be feasible and worthwhile, even though the effort involved in organization of and participation in community based evaluations in the area is substantial.

The conceptual model for SWS technology evaluation was validated with respect to its comprehensiveness and wellfoundedness. The latter was assessed by verifying the correctness of the methodological approach that was followed to derive the model. Comprehensiveness was assessed by discussing the completeness of the framework and its practical applicability to relate different SWS evaluations to each other.

The contributions towards better SWS test data were validated by assessing the provided portal against the engineering objectives that motivated their design. It was found that the portal achieves its goals. This claim is additionally supported through the illustration of its usability in the creation of the Jena Geography Dataset and through its acceptance and usage by the wider community.

Finally, the two benchmarks contributed by this thesis were each meta-evaluated by four means. First, it was assessed that they meet the design objectives that motivated their creation. Second, they were formally meta-evaluated against the evaluation requirements defined as part of the conceptual model. Third, the proper dissemination of the benchmarks was explained and fourth, the strengths and weaknesses of the benchmarks were discussed on a more abstract level with a focus on general lessons learned. Both benchmarks were found to meet their design objectives and achieve all evaluation requirements, most of them without restrictions. Additionally, the successful application of the benchmarks and the positive feedback by participants in the corresponding benchmarking events illustrate the acceptance of the benchmarks by the community.

Having summarized the thesis, we now discuss areas of possible future work basing on the results from this thesis.

9.2. Future Work

The importance of continuous benchmarking was stressed several times throughout this thesis. Benchmarking in an area of research should only come to an end if no more progress is being made in that area. Until this point, a scientific area and its benchmarks should co-evolve. This implies that during this process, evaluations should be performed on a regular basis. The most obvious and important future work is thus the continuation of the benchmarking events carried out as part of this thesis work, be it in their current, or some altered and improved form.

Apart from this general remark, it should be noted that benchmarking of SWS still needs to be considered in its beginnings in many ways. This thesis laid foundation for a more systematic and thorough treatment of the issue, but could not even attempt providing a definite solution to all benchmarking needs in the area. Furthermore, the reliability of evaluations can only be assessed in a definite way if they can be compared to alternative evaluations and, even better, if they are shown to make the right predictions and lead to the desired scientific progress. This is only feasible as a long term process beyond the scope of one thesis. Feitelson discusses the temptation of fast and easy measurement solutions and the difficulty of defining and designing truly good metrics:

> "Of course, coming up with good metrics is not easy. One should especially beware of the temptation of measuring what is easily accessible, and using it as a proxy for what is really required. Baseball statistics provide an illuminating example in this respect. Players were (and still are) often evaluated by their batting average and how fast they can run, and pitchers by how fast they can throw the ball. But as it turns out, these metrics don't correlate with having a positive effect on winning baseball games. Therefore other metrics are needed. What metrics are the most effective is determined by experimentation: when you have a candidate metric, try it out and see if it makes the right predictions. After years of checking vast amounts of data by many people, simple and effective metrics can be distilled. In the case of baseball, the metric for hitters is their on-base percentage; for pitchers it is hitters struck out and home runs allowed." [Fei06]

This nicely illustrates the general need for continuous research on evaluations in general and metrics in particular, also in the area of SWS. Apart from this principle necessity, a list of more concrete suggestions for future work is provided in the following. The suggestions are classified into improvements of the contributions made by this thesis and complementary future work.

9.2.1. Possible Improvements of the Thesis Contributions

The conceptual model for SWS technology evaluation is considered to be comprehensive and final as is. However, a number of further improvements of the three concrete benchmarking contributions are possible.

Test Data for SWS Evaluation

With respect to test data for SWS evaluation, the basic principle *the more, the better* holds. Much more data than is currently available is clearly desirable. In concrete terms, the Jena Geography Dataset could be extended regarding the issues discussed in Section 5.4. First of all, the JGD is of a limited size. A size larger than 200 services would generally enable more reliable measurements.

More specifically, extension in two directions would be beneficial. For one thing, the JGD is limited to a single domain (geography, geocoding). Extending it to other domains would result in a higher diversity of services and thus more representative evaluation results. For another thing, the JGD currently comprises exclusively web-safe data services, i.e., services which provide information or offer computations but do not create lasting world-altering effects. Complementing the collection with such services would allow a more comprehensive coverage of the problem space.

Furthermore, semantic descriptions for the JGD services are currently only available for a subset of 50 services in four different description approaches. Descriptions for the full data set, complementary descriptions in more formalisms, including in particular standard WSMO/WSML and OWL-S notation and descriptions comprising fully specified preconditions and effects would be highly desirable.

Apart from the concrete test data available, improvements of the OPOSSum portal are also possible. In some ways, OPOSSum needs to be considered a prototypical implementation. The usability of the interface could be improved in a reimplementation that leverages modern Web technologies like JavaScript and Ajax. This might also lead to still greater acceptance and usage of the portal by the community.

Also some of the more advanced functionality in the portal has been implemented in a limited way. This particularly regards the ability to provide extensive analysis of relevance judgments consistency. Furthermore, improvements of the search function are conceivable and a version management of the listed data would help keeping track of changes in the listed data. Finally, the integration of OPOSSum with existing tools like the SME2 evaluation environment could be improved and the storage of large datasets in OPOSSum be supported in a better automated way. This also includes the capability of letting users choose the amount of metadata and structure to provide more freely.

Benchmarking the Functional Scope of SWS Frameworks

The functional scope benchmark could be extended towards more scenarios. Such extension might cover new functional challenges, some of which are already envisioned: services with optional inputs and defaults, highly configurable services or scenarios that involve the conversion of units. It would also be worthwhile to investigate whether scenarios from use case based project evaluations (see Section 3.1.4) which have meanwhile been performed could be meaningfully integrated.

For the development of new scenarios as well as possible improvements of the existing ones lessons learned so far should be considered. Scenarios should strive for a clear separation of different functional challenges that allows addressing them as independently as possible. Additionally, the specification of success criteria for acceptable solutions in a clear, rigid and unambiguous manner is highly recommended. Finally, a higher degree of automation of the correctness verification of solutions would be desirable.

Regarding the benchmarking methodology and evaluation process, more emphasis on shared, properly documented and reusable solutions is suggested. Besides, a better, more comprehensive documentation of the evaluation taking place at each evaluation workshop would promote transparency and support easier meta-evaluations of the process. Finally, more specific and formal agreements about the commitments and responsibilities of the people involved, organizers as well as participants, may further promote a professionally performed evaluation.

On a more general level, it might be useful to also work towards means for reliably assessing the flexibility of solutions and the effort involved in creating them. Even though previous attempts in this direction showed the difficulties involved in this endeavor, such assessment would provide an important context for the certification of functional capabilities. With frameworks and tools becoming more mature over time, assessing their usability and effectiveness (in terms of programmer productivity) might also become more feasible than it used to be.

Benchmarking SWS Matchmaking

The work on benchmarking SWS matchmakers can be extended primarily in two directions. For one thing, the evaluation should be repeated on a bigger scale, for another thing, comprehensive and detailed analyses of the evaluation results are desirable.

As discussed in Section 7.8, the evaluation had to be performed on a downsized version of the Jena Geography Dataset in order to avoid overcharging participants. For improved reliability of results and to produce more data for an in-depth analysis, the evaluation should be repeated with more descriptions. An obvious first step would be to use the full JGD, further extensions follow the directions for more test

data outlined above in Section 9.2.1. Besides, higher participation in terms of more matchmakers, a still greater variety of formalisms and different sets of descriptions for each matchmaker (in the current edition, the SAWSDL matchmakers relied on the same set of descriptions) would also foster a true cross evaluation and provide a bigger basis for in-depth analyses of the causes of observed performance results.

Such thorough analyses are best performed by participants in the evaluation who possess the necessary intimate knowledge of the evaluated approaches. The mentioned analyses had not yet been available at the time of writing of this thesis (end of 2009). However, a book project soliciting corresponding chapters by participants is in planning. It is expected that these analyses will allow tracing causes for performance characteristics to answer fundamental questions about SWS matchmaking. These concern the level of information that needs to be shared between requesters and providers, the most appropriate detailedness of service descriptions, the best patterns and formalisms to describe services or the properties that make certain retrieval problems challenging. Such insights can only be distilled in a definite way over time from repeated executions of the benchmark in combination with the mentioned in-depth analyses. However, answers to these questions, even preliminary ones, are most helpful in order to advance SWS matchmaking research and thus the most important area of future work regarding this benchmark.

On a practical level, the desired analyses could be encouraged by improving the available tool support for visualizing and tracing retrieval correctness results. Corresponding work is already in progress for the SME2 evaluation environment.

9.2.2. Complementary Future Work

Apart from the future work discussed above, which concerns immediate confined improvements of the thesis contributions, more substantial future work that is complementary to the thesis contributions is possible. This concerns primarily the development of benchmarks for the tasks and criteria not covered by this thesis. The scope of the concrete benchmarking contributions of this thesis was defined in Section 4.5. Two extensions which are complementary to this scope are particularly desirable, namely, an extension of the scope towards service composition and towards more extensive usability evaluations.

This thesis focused on service discovery and covered service composition only in a basic form to a very limited extent. However, besides service discovery, automated service composition is the primary use case motivating SWS. Thus, benchmarks directly focusing on service composition covering more complex use cases than those covered by the composition tasks of the functional scope benchmark are clearly desirable. Such benchmarks should support an evaluation of the capabilities of composition approaches similar to the functional scope benchmark presented in this thesis, an evaluation of the runtime performance of service composition planners

but also a more general assessment of the extent to which semantics may ease the process of composing services. The work performed in the mediation track of the SWS Challenge, the service composition performance evaluation done in the WS Challenge and the International Planning Competition from the AI community may provide starting points and important inputs to the further development of such benchmarks.

With respect to criteria, usability evaluations are expected to become more and more important. The benchmarks contributed by this thesis focused primarily on a technical level. They are designed to support researchers and developers in investigating specific properties of their technologies. With technologies becoming more mature, evaluations of their usability need to receive more attention. Such evaluations concern the tool support available for a certain technology, its complexity and thus ease of learning and ease of use and ultimately, the competitive advantage that is achieved by using it compared to using established technology.

SWS are expected to make service oriented computing more efficient. This is basically a software engineering claim [Pet06]. In the long run, SWS benchmarks do not only need to support a meaningful comparison of different SWS technologies, they should also allow investigating whether this basic assumption behind SWS is valid. Such evaluations are extremely difficult, which is illustrated by the corresponding efforts within the SWS Challenge (see Section 8.6.4). Yet, means to assess this claim will most likely be needed before SWS will be employed on a large scale.

9.3. Conclusion

More than a century ago, Lord Kelvin has articulated the need for measurement in any scientific area in clear-cut words when remarking: "When you can measure what you are speaking about and express it in numbers, you know something about it; but when you cannot measure it, when you cannot express it in numbers, your knowledge is of a meager and unsatisfactory kind."

This thesis provided a starting point towards making the characteristics of SWS technology measurable. We thus hope that the thesis contributions help people to understand SWS technology better, to indeed learn something about it such that further scientific development is promoted.

In the motivation of this thesis, we argued that there is a need for more experimentation and more measurement in computer science. Feitelson remarks about this:

> "We also don't really know how to measure the quantity, quality, or complexity of software, or the productivity of software production, the performance of microprocessors or supercomputers, or the reliability or availability of distributed systems, to mention but a few. It's not that

> no metric is available. It's that the suggested metrics all have obvious deficiencies, none are widely used, and that there is relatively little discussion about how to improve them." [Fei06]

We hope that this thesis will serve as the foundation for more future discussion about how the metrics and measures to assess and compare SWS technology can be further improved. We also hope that this thesis will generally stimulate more experimentation and measurement in the area and thus come to productive use for the advancement of SWS technology.

References

[ÅÅLS06] Cécile Åberg, Johan Åberg, Patrick Lambrix, and Nahid Shahmehri. A platform to evaluate the technology for service discovery in the semantic web. In *Proceedings of the Twenty-First National Conference on Artificial Intelligence (AAAI06)*, Boston, Massachusetts, USA, July 2006.

[Åbe07] Cécile Åberg. *An Evaluation Platform for Semantic Web Technology*. PhD thesis, Department of Computer and Information Science, Linköpings Universitet Sweden, 2007.

[AGY05] Paolo Avesani, Fausto Giunchiglia, and Mikalai Yatskevich. A large scale taxonomy mapping evaluation. In *Proceedings of the 4th International Semantic Web Conference (ISWC2005)*, pages 67–81, Galway, Ireland, November 2005.

[AHKZ08] Witold Abramowicz, Konstanty Haniewicz, Monika Kaczmarek, and Dominik Zyskowski. E-marketplace for semantic web services. In *Proceedings of the 6th International Conference on Service-Oriented Computing (ICSOC08)*, pages 271–285, Sydney, Australia, December 2008.

[AL05] Sudhir Agarwal and Steffen Lamparter. Smart - a semantic matchmaking portal for electronic markets. In *Proceedings of 7th IEEE International Conference on E-Commerce Technology (CEC 2005)*, pages 405–408, München, Germany, July 2005.

[AvH04] Grigoris Antoniou and Frank van Harmelen. *A Semantic Web Primer*. MIT Press, 2004.

[Bas92] Victor R. Basili. Software modeling and measurement: the goal/question/metric paradigm. Technical report, University of Maryland at College Park, College Park, MD, USA, 1992.

[BB93] Richard Bache and Gualtiero Bazzana. *Software Metrics for Product Assessment*. McGraw-Hill Book Company Europe, 1993.

[BBK+08] Ajay Bansal, M. Brian Blake, Srividya Kona, Steffen Bleul, Thomas Weise, and Michael C. Jaeger. WSC-08: Continuing the web services challenge. In *Proceedings of the 10th IEEE International Conference on E-Commerce Technology (CEC2008) / 5th IEEE International Conference on Enterprise Computing, E-Commerce and E-Services (EEE2008)*, pages 351–354, Washington, DC, USA, 2008.

[BCC+06a] Marco Brambilla, Irene Celino, Stefano Ceri, Dario Cerizza, Emanuele Della Valle, Federico Facca, and Christina Tziviskou. Improvements and future perspectives on web engineering methods for automating web services mediation, choreography and discovery: SWS-Challenge phase III. In *Third Workshop of the Semantic Web Service Challenge 2006 - Challenge on Automating Web Services Mediation, Choreography and Discovery*, Athens, GA, USA, November 2006.

[BCC+06b] Marco Brambilla, Stefano Ceri, Dario Cerizza, Emanuele Della Valle, Federico Facca, and Christina Tziviskou. Coping with requirements changes: SWS-Challenge phase II. In *Second Workshop of the Semantic Web Service Challenge 2006 - Challenge on Automating Web Services Mediation, Choreography and Discovery*, Budva, Montenegro, June 2006.

[BCJW06] M. Brian Blake, William Cheung, Michael C. Jaeger, and Andreas Wombacher. WSC-06: the web service challenge. In *Proceedings of the Eighth IEEE International Conference on E-Commerce Technology (CEC 2006) and Third IEEE International Conference on Enterprise Computing, E-Commerce and E-Services (EEE 2006)*, Palo Alto, California, USA, June 2006.

[BCJW07] M. Brian Blake, William Kwok-Wai Cheung, Michael C. Jaeger, and Andreas Wombacher. WSC-07: evolving the web services challenge. In *Proceedings of the 9th IEEE International Conference on E-Commerce Technology (CEC 2007)*, Tokyo, Japan, July 2007.

[BCR94] Victor R. Basili, Gianluigi Caldiera, and H. Dieter Rombach. The goal question metric approach. In *Encyclopedia of Software Engineering*, pages 528–532. John Wiley and Sons, Inc., 1994.

[BCV+08] Marco Brambilla, Stefano Ceri, Emanuele Della Valle, Federico M. Facca, and Christina Tziviskou. A software engineering approach

based on WebML and BPMN to the mediation scenario of the sws challenge. In Charles Petrie, Holger Lausen, Michal Zaremba, and Tiziana Margaria, editors, *Semantic Web Service Challenge — Results from the First Year*. Springer, 2008.

[Bey03] Wolfgang Beywl. Selected comments to the standards for evaluation of the german evaluation society – english edition –. Technical report, German Evaluation Society (DeGEval), 2003.

[BL98] Tim Berners-Lee. Semantic web road map. available online at http://www.w3.org/DesignIssues/Semantic.html, September 1998.

[BLHL01] Tim Berners-Lee, James Hendler, and Ora Lassila. The semantic web. *Scientific American*, 5, May 2001.

[BLW04] Steffen Balzer, Thorsten Liebig, and Matthias Wagner. Pitfalls of owl-s: a practical semantic web use case. In *Proceedings of the Second International Conference on Service-Oriented Computing (ICSOC2004)*, New York, NY, USA, November 2004.

[BOI09] Ayse B. Bener, Volkan Ozadali, and Erdem Savas Ilhan. Semantic matchmaker with precondition and effect matching using SWRL. *Expert Systems with Applications*, 36(5):9371 – 9377, 2009.

[Bro08] Michael L. Brodie. Foreword. In Charles Petrie, Holger Lausen, Michal Zaremba, and Tiziana Margaria, editors, *Semantic Web Service Challenge — Results from the First Year*. Springer, 2008.

[BS09] Christian Bizer and Andreas Schultz. The Berlin SPARQL Benchmark. *International Journal on Semantic Web and Information Systems*, 5(2):1–24, 2009.

[BSR+05] Christopher J. C. Burges, Tal Shaked, Erin Renshaw, Ari Lazier, Matt Deeds, Nicole Hamilton, and Gregory N. Hullender. Learning to rank using gradient descent. In *Proceedings of the Twenty-Second International Conference on Machine Learning (ICML05)*, pages 89–96, Bonn, Germany, August 2005.

[BTW05] M. Brian Blake, Kwok Ching Tsui, and Andreas Wombacher. The EEE-05 Challenge: a new web service discovery and composition competition. In *Proceedings of the 2005 IEEE International Conference on e-Technology, e-Commerce, and e-Services (EEE 2005)*, Hong Kong, China, March 2005.

[BWG09] Steffen Bleul, Thomas Weise, and Kurt Geihs. The web service challenge - a review on semantic web service composition. In *Proceedings of the Workshop on Service-Oriented Computing at KIVS 2009*, Kassel, Germany, March 2009.

[BYRN99] Ricardo Baeza-Yates and Berthier Ribeiro-Neto. *Modern Information Retrieval*. Addison Wesley, 1999.

[BZ07] Victor R. Basili and Marvin V. Zelkowitz. Empirical studies to build a science of computer science. *Communications of the ACM*, 50(11):33–37, 2007.

[CCC+08] Alessio Carenini, Dario Cerizza, Marco Comerio, Emanuele Della Valle, Flavio De Paoli, Andrea Maurino, Matteo Palmonari, Matteo Sassi, and Andrea Turati. Semantic web service discovery and selection: a test bed scenario. In *In Proceedings of the 6th International Workshop on Evaluation of Ontology-based Tools and the Semantic Web Service Challenge (EON-SWSC08)*, Tenerife, Canary Islands, Spain, June 2008.

[CDG+06] Liliana Cabral, John Domingue, Stefania Galizia, Alessio Gugliotta, Vlad Tanasescu, Carlos Pedrinaci, and Barry Norton. Irs-iii: A broker for semantic web services based applications. In *Proceedings of the 5th International Semantic Web Conference (ISWC 2006)*, Athens, GA, USA, November 2006.

[CLC+04] José Manuel López Cobo, Silvestre Losada, Óscar Corcho, V. Richard Benjamins, Marcos Niño, and Jesús Contreras. Sws for financial overdrawn alerting. In *Proceedings of the Third International Semantic Web Conference (ISWC2004)*, Hiroshima, Japan, November 2004.

[CNS+05] Simona Colucci, Tommaso Di Noia, Eugenio Di Sciascio, Francesco M. Donini, and Marina Mongiello. Concept abduction and contraction for semantic-based discovery of matches and negotiation spaces in an e-marketplace. *Electronic Commerce Research and Applications*, 4(4):345–361, 2005.

[Den05] Peter J. Denning. Is computer science science? *Communications of the ACM (CACM)*, 48(4):27–31, 2005.

[dFEJ+09] David de Francisco, Mark Evenson, Zlatina Jordanova, Ewelina Szczekocka, Rasha Mozil, Maciej Zaremba, Bernhard Schreder, Jesus Contreras, and Ivan Martinez. SUPER Deliverable 8.6: YATOSP

demonstrator and telco use case metrics analysis. Technical report, Project IST 026850 SUPER, March 2009.

[DFLO06a] Christian Drumm, Andreas Friesen, Jens Lemcke, and Daniel Oberle. WP5: service mediation, D5.7b: final prototype of mediation and discovery in a real world scenario. Technical report, DIP Project FP6 - 507483, November 2006.

[DFLO06b] Christian Drumm, Andreas Friesen, Jens Lemcke, and Daniel Oberle. WP5: service mediation, D5.8b: final prototype of mediation and composition in a real world scenario. Technical report, DIP Project FP6 - 507483, November 2006.

[DHL+07] Philippe Duchesne, Joerg Hoffmann, Joel Langlois, François Tertre, Jochen Bercker, and Andor Lips. SWING Deliverable 1.1: Use case definition and I&T requirements. Technical report, FP6 - 26514 SWING, Febuary 2007.

[DHM+04] Xin Dong, Alon Y. Halevy, Jayant Madhavan, Ema Nemes, and Jun Zhang. Similarity search for web services. In *Proceedings of the Thirtieth International Conference on Very Large Data Bases (VLDB2004)*, pages 372–383, Toronto, Canada, August, September 2004.

[DM06] Gianluca Demartini and Stefano Mizzaro. A classification of IR effectiveness metrics. In *Proceedings of the 28th European Conference on IR Research (ECIR06)*, pages 488–491, London, UK, April 2006.

[dSFE07] Claudia d'Amato, Steffen Staab, Nicola Fanizzi, and Floriana Esposito. Efficient discovery of services specified in description logics languages. In *Proceedings of the First International Joint Workshop SMR^2 on Service Matchmaking and Resource Retrieval in the Semantic Web at the 6th International Semantic Web Conference(ISWC2007)*, Busan, South Korea, November 2007.

[ea85] Anon et al. A measure of transaction processing power. *Datamation*, 31(7), 1985.

[Erl05] Thomas Erl. *Service-Oriented Architectures - Concepts, Technology, and Design*. Pearson Education, 2005.

[ES07] Jérôme Euzenat and Pavel Shvaiko. *Ontology Matching*. Springer, 2007.

[Fei06] Dror G. Feitelson. Experimental computer science: The need for a cultural change. Manuscript, online at http://www.cs.huji.ac.il/∼feit/papers/exp05.pdf, December 2006.

[Fei07] Dror G. Feitelson. Introduction. *Communications of the ACM*, 50(11):24–26, 2007.

[Fen91] Norman E. Fenton. *Software Metrics - A rigorous approach*. Chapman & Hall, 1991.

[FG05] Andreas Friesen and Stephan Grimm. DIP Deliverable D4.8: Discovery specification. Technical report, DIP, Data, Information and Process Integration with Semantic Web Services, FP6 - 507483, December 2005.

[FHL05] Dieter Fensel, James A. Hendler, and Henry Lieberman, editors. *Spinning the Semantic Web: Bringing the World Wide Web to Its Full Potential*. MIT Press, 2005.

[Fie00] Roy Thomas Fielding. *Architectural Styles and the Design of Network-based Software Architectures*. PhD thesis, University of California, Irvine, CA, USA, 2000.

[Fin82] L. Finkelstein. What is not measurable, make measurable. *Measurement and Control*, 15, 1982.

[Fin84] L. Finkelstein. A review of the fundamental concepts of measurement. *Measurement*, 2(1), 1984.

[Fis05] Thomas Fischer. Entwicklung einer Evaluationsmethodik für Semantic Web Services und Anwendung auf die DIANE Service Descriptions (in German). Master's thesis, Institute for Program Structures and Data Organization, University Karlsruhe, August 2005.

[FKL+05] Dieter Fensel, Uwe Keller, Holger Lausen, Axel Polleres, and Ioan Toma. WWW or what is wrong with web service discovery. In *W3C Workshop on Frameworks for Semantics in Web Services*, Innsbruck, Austria, June 2005.

[FMS+07] Nadine Fröhlich, Thorsten Möller, Heiko Schuldt, António Lopes, Matthias Klusch, Ari Kinnunen, Alexandre de Oliveira e Sousa, Federico Bergenti, Danilo Bonardi, Heimo Laamanen, Mihael Cankar, and Matteo Vasirani. CASCOM Deliverable D7.2: Validation and trial results. Technical report, CASCOM Project FP6 - 511632, December 2007.

[FN06] Andreas Friesen and Kioumars Namiri. Towards semantic service selection for B2B integration. In *Proceedings of the Joint workshop on web services modeling and implementation using sound web engineering practices and methods, architectures and technologies for e-service engineering (SMIWEP-MATeS'06) at the Sixth International Conference on Web Engineering (ICWE06)*, Palo Alto, CA, USA, July 2006.

[FPG94] Norman Fenton, Shari Lawrence Pfleeger, and Robert L. Glass. Science and substance: A challenge to software engineers. *IEEE Software*, 11(4):86–95, 1994.

[FSM+07] Nadine Fröhlich, Heiko Schuldt, Thorsten Möller, Ari Kinnunen, Federico Bergenti, and Heimo Laamanen. CASCOM Deliverable D7.1: Validation and trial plan. Technical report, CASCOM Project FP6 - 511632, Febuary 2007.

[GA95] Ran Giladi and Niv Ahituv. SPEC as a performance evaluation measure. *IEEE Computer*, 28(8):33–42, 1995.

[GANJ06] Yasser Ganjisaffar, Hassan Abolhassani, Mahmood Neshati, and Mohsen Jamali. A similarity measure for OWL-S annotated web services. In *Proceedings of the 2006 IEEE/WIC/ACM International Conference on Web Intelligence (WI2006)*, pages 621–624, Hong Kong, December 2006.

[GC05] Raúl García-Castro. Knowledge Web Deliverable D2.1.4: Specification of a methodology, general criteria, and benchmark suites for benchmarking ontology tools. Technical Report KWEB/2004/D2.1.4/v1.5, KWEB EU-IST-2004-507482, February 2005.

[GC07] Raúl García-Castro. Defining benchmark suites with reusability in mind. online available at `http://km.aifb.uni-karlsruhe.de/ws/eon2007/Garcia-Castro2007_-_Reusable_benchmark_suites.pdf`, 2007.

[GC08] Raúl García-Castro. *Benchmarking Semantic Web Technology*. PhD thesis, Universidad Politécnica de Madrid - Facultad de Informática, 2008.

[GCGP05] Raul Garcia-Castro and Asunción Gómez-Pérez. Guidelines for benchmarking the performance of ontology management APIs. In *Proceedings of the 4th International Semantic Web Conference (ISWC2005)*, pages 277–292, Galway, Ireland, November 2005.

[GHD02] Günther Gediga, Kai-Christoph Hamborg, and Ivo Düntsch. Evaluation of software systems. In Allen Kent and James G. Williams, editors, *Encyclopedia of Computer Science and Technology*, volume 45, pages 127 – 153. CRC, 2002.

[GMP04] Stephan Grimm, Boris Motik, and Chris Preist. Variance in e-business service discovery. In *Proceedings of the ISWC 2004 Workshop on Semantic Web Services: Preparing to Meet the World of Business Applications*, Hiroshima, Japan, November 2004.

[GMP06] Stephan Grimm, Boris Motik, and Chris Preist. Matching semantic service descriptions with local closed-world reasoning. In *Proceedings of the 3rd European Semantic Web Conference (ESWC 2006)*, Budva, Montenegro, June 2006.

[GP04] Asunción Gómez-Pérez. Ontology evaluation. In Steffen Staab and Rudi Studer, editors, *Handbook on Ontologies*. Springer, 2004.

[GPH05] Yuanbo Guo, Zhengxiang Pan, and Jeff Heflin. LUBM: a benchmark for OWL knowledge base systems. *Journal on Web Semantics*, 3(2-3):158–182, 2005.

[GQPH07] Yuanbo Guo, Abir Qasem, Zhengxiang Pan, and Jeff Heflin. A requirements driven framework for benchmarking semantic web knowledge base systems. *IEEE Transactions on Knowledge and Data Engineering*, 19(2):297–309, 2007.

[GTD+06] Alessio Gugliotta, Vlad Tanasescu, John Domingue, Rob Davies, Leticia Gutiérrez-Villarías, Mary Rowlatt, Marc Richardson, and Sandra Stinčić. Benefits and challenges of applying semantic web services in the e-government domain. In *Semantics 2006*, Vienna, Austria, November 2006.

[GTD+07] Alessio Gugliotta, Vlad Tanasescu, John Domingue, Leticia Gutierrez, Rob Davies, Mary Rowlatt, Jon Bryant, Sandra Stincic, and Marc Richardson. WP 9: Case Study eGovernment, D9.14: SWS enhanced GIS prototype (IRSIII) final version. Technical report, DIP Project FP6 - 507483, January 2007.

[Gul56] C. D. Gull. Seven years of work on the organization of materials in the special library. *American Documentation*, 7(4):320–329, 1956.

[Har92] Donna Harman. Overview of the first Text REtrieval Conference (TREC-1). In *Proceedings of the first Text REtrieval Conference (TREC-1)*, Gaithersbury, MD, USA, November 1992.

[Har93] Juris Hartmanis. Some observations about the nature of computer science. In *Proceedings of the 13th Conference on Foundations of Software Technology and Theoretical Computer Science*, pages 1–12, 1993.

[HBG+98] Andy Hoetzel, Alain Benhaim, Nigel Griffiths, Chet Holliday, and Norbert Pistoor. *Benchmarking in Focus.* IBM Redbooks, 1998.

[Hen00] John L. Henning. SPEC CPU2000: Measuring CPU performance in the new millennium. *IEEE Computer*, 33(7):28–35, 2000.

[Hen01] James A. Hendler. Agents and the semantic web. *IEEE Intelligent Systems*, 16(2):30–37, 2001.

[HJK04] Andreas Heß, Eddie Johnston, and Nicholas Kushmerick. ASSAM: a tool for semi-automatically annotating semantic web services. In *Proceedings of the 3rd International Semantic Web Conference (ISWC04)*, pages 320–334, Hiroshima, Japan, November 2004.

[HKRK07] Mohamed Hamdy, Birgitta König-Ries, and Ulrich Küster. Non-functional parameters as first class citizens in service description and matchmaking — an integrated approach. In *Proceedings of the first International Workshop on Non Functional Properties and Service Level Agreements in Service Oriented Computing (NFPSLA-SOC 2007)*, Vienna, Austria, September 2007.

[HKZ08] Konstanty Haniewicz, Monika Kaczmarek, and Dominik Zyskowski. Semantic web services applications — a reality check. *Wirtschaftsinformatik*, 1:39–46, 2008.

[Hof08] Jörg Hoffmann. SWING Deliverable 2.4: Semantic web geoprocessing services. Technical report, FP6 - 26514 SWING, January 2008.

[HRK09] Pascal Hitzler, Sebastian Rudolph, and Markus Krötzsch. *Foundations of Semantic Web Technologies.* Chapman & Hall/Crc, 2009.

[HT06] Andreas Höfer and Walter F. Tichy. Status of empirical research in software engineering. In *Revised Papers of the International Dagstuhl Workshop on Empirical Software Engineering Issues. Critical Assessment and Future Directions*, pages 10–19, Dagstuhl Castle, Germany, June 2006. Springer.

[ING+06] Kashif Iqbal, Sanaullah Nazir, Sven Groppe, Adina Sirbu, and Pasi Titinen. Adaptive Services Grid Deliverable D1.I-3: Service match-

maker and query processor – 1st release. Technical report, Adaptive Services Grid Project FP6 - 004617, May 2006.

[INS07] Kashif Iqbal, Sanaullah Nazir, and Adina Sirbu. Adaptive Services Grid Deliverable D1.I-4: Service matchmaker and query processor – 2nd release. Technical report, Adaptive Services Grid Project FP6 - 004617, March 2007.

[JK02] Kalervo Järvelin and Jaana Kekäläinen. Cumulated gain-based evaluation of IR techniques. *ACM Transactions on Information Systems*, 20(4):422–446, 2002.

[Jon00] Karen Sparck Jones. Further reflections on TREC. *Information Processing and Management: an International Journal*, 36(1):37–85, January 2000.

[Jon05] Karen Sparck Jones. Meta-reflections on TREC. In *TREC Experiment and Evaluation in Information Retrieval*. MIT Press, 2005.

[JRGL+05] Michael C. Jaeger, Gregor Rojec-Goldmann, Christoph Liebetruth, Gero Mühl, and Kurt Geihs. Ranked matching for service descriptions using OWL-S. In *Kommunikation in Verteilten Systemen (KiVS), 14. ITG/GI-Fachtagung Kommunikation in Verteilten Systemen (KiVS 2005)*, pages 91–102, Kaiserslautern, Germany, March 2005.

[KB08] Christoph Kiefer and Abraham Bernstein. The creation and evaluation of iSPARQL strategies for matchmaking. In *Proceedings of the 5th European Semantic Web Conference (ESWC2008)*, pages 463–477, Tenerife, Canary Islands, Spain, June 2008.

[KBB+09] Srividya Kona, Anjay Basal, M. Brian Blake, Steffen Bleul, and Thomas Weise. WSC-09: a quality of service-oriented web service challenge. In *Proceedings of IEEE Joint Conference on E-Commerce Technology and Enterprise Computing, E-Commerce and E-Services (CEC/EEE 2009)*, Vienna, Austria, July 2009.

[KFK07] Mahboob Alam Khalid, Benedikt Fries, and Patrick Kapahnke. OWLS-TC - OWL-S service retrieval test collection version 2.2 user manual. online at http://www.semwebcentral.org/projects/owls-tc/, November 2007.

[KFKK08] Mahboob Alam Khalid, Benedikt Fries, Patrick Kapahnke, and Matthias Klusch. SAWSDL service retrieval test collection. online at http://www.semwebcentral.org/projects/sawsdl-tc/, July 2008.

[KFKS05] Matthias Klusch, Benedikt Fries, Mahboob Khalid, and Katia Sycara. OWLS-MX: Hybrid OWL-S service matchmaking. In *Proceedings of the First International AAAI Fall Symposium on Agents and the Semantic Web*, Arlington, Vriginia, USA, November 2005.

[KFS06] Matthias Klusch, Benedikt Fries, and Katia Sycara. Automated semantic web service discovery with OWLS-MX. In *Proceedings of the 5th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2006)*, Hakodate, Japan, May 2006.

[KG90] Maurice Kendall and Jean Dickinson Gibbons. *Rank Correlation Methods*. Oxford University Press, New York, fifth edition edition, 1990.

[Kis05] Kazuaki Kishida. Property of average precision and its generalization: An examination of evaluation indicator for information retrieval experiments. Technical Report NII-2005-014E, National Institute of Informatics, Tokyo, Japan, October 2005.

[Kit96] Barbara Ann Kitchenham. Evaluating software engineering methods and tool part 1: The evaluation context and evaluation methods. *SIGSOFT Software Engineering Notes*, 21(1):11–14, 1996.

[KJ02] Jaana Kekäläinen and Kalervo Järvelin. Using graded relevance assessments in IR evaluation. *Journal of the American Society for Information Science and Technology*, 53(13):1120–1129, 2002.

[KK07] Frank Kaufer and Matthias Klusch. Performance of hybrid WSML service matching with WSMO-MX: Preliminary results. In *Proceedings of the First International Joint Workshop SMR^2 on Service Matchmaking and Resource Retrieval in the Semantic Web at the 6th International Semantic Web Conference(ISWC2007)*, Busan, South Korea, November 2007.

[KK08] Matthias Klusch and Patrick Kapahnke. Semantic web service selection with SAWSDL-MX. In *Proceedings of the 2nd International Workshop SMR^2 on Service Matchmaking and Resource Retrieval in the Semantic Web at the 7th International Semantic Web Conference (ISWC08)*, Karlsruhe, Germany, October 2008.

[KKF08] Matthias Klusch, Patrick Kapahnke, and Benedikt Fries. Hybrid semantic web service retrieval: A case study with OWLS-MX. In *Proceedings of the 2nd IEEE International Conference on Semantic Computing (ICSC2008)*, Santa Clara, CA, USA, August 2008.

[KKK+09] Matthias Klusch, Mahboob Alam Khalid, Patrick Kapahnke, Benedikt Fries, and Martin Vasileski. OWLS-TC - OWL-S service retrieval test collection version 3 user manual. online at http://www.semwebcentral.org/projects/owls-tc/, June 2009.

[KKP08] R. Knackstedt, D. Kuropka, and O. Polyvyanyy. An ontology-based service discovery approach for the provisioning of product-service bundles. In *Proceedings of the 16th European Conference on Information Systems (ECIS08)*, Galway, Ireland, June 2008.

[KKR06a] Ulrich Küster and Birgitta König-Ries. Discovery and mediation using DIANE service descriptions. In *Third Workshop of the Semantic Web Service Challenge 2006 - Challenge on Automating Web Services Mediation, Choreography and Discovery*, Athens, GA, USA, November 2006.

[KKR06b] Ulrich Küster and Birgitta König-Ries. Dynamic binding for BPEL processes — a lightweight approach to integrate semantics into web services. In *Second International Workshop on Engineering Service-Oriented Applications: Design and Composition (WESOA06) at 4th International Conference on Service Oriented Computing (ICSOC06)*, Chicago, Illinois, USA, December 2006.

[KKR07a] Ulrich Küster and Birgitta König-Ries. Semantic mediation between business partners — a SWS-Challenge solution using DIANE service descriptions. In *Proceedings of the International Workshop on Service Composition & SWS Challenge at the 2007 IEEE/WIC/ACM International Conference on Web Intelligence (WI 2007)*, Silicon Valley, USA, November 2007.

[KKR07b] Ulrich Küster and Birgitta König-Ries. Semantic service discovery with DIANE service descriptions. In *Proceedings of the International Workshop on Service Composition & SWS Challenge at the 2007 IEEE/WIC/ACM International Conference on Web Intelligence (WI 2007)*, Silicon Valley, USA, November 2007.

[KKR07c] Ulrich Küster and Birgitta König-Ries. Service discovery using diane service descriptions — a solution to the SWS-Challenge discovery scenarios. In *Fourth Workshop of the Semantic Web Service Challenge - Challenge on Automating Web Services Mediation, Choreography and Discovery*, Innsbruck, Austria, June 2007.

[KKR07d] Ulrich Küster and Birgitta König-Ries. Supporting dynamics in service descriptions - the key to automatic service usage. In *Proceedings*

of the Fifth International Conference on Service Oriented Computing (ICSOC07), Vienna, Austria, September 2007.

[KKR08a] Ulrich Küster and Birgitta König-Ries. Evaluating semantic web service matchmaking effectiveness based on graded relevance. In *Proceedings of the 2nd International Workshop SMR^2 on Service Matchmaking and Resource Retrieval in the Semantic Web at the 7th International Semantic Web Conference (ISWC08)*, Karlsruhe, Germany, October 2008.

[KKR08b] Ulrich Küster and Birgitta König-Ries. On the empirical evaluation of semantic web service approaches: Towards common SWS test collections. In *Proceedings of the 2nd IEEE International Conference on Semantic Computing (ICSC2008)*, Santa Clara, CA, USA, August 2008.

[KKR08c] Ulrich Küster and Birgitta König-Ries. Semantic service discovery with DIANE service descriptions. In Charles Petrie, Holger Lausen, Michal Zaremba, and Tiziana Margaria, editors, *Semantic Web Service Challenge — Results from the First Year*. Springer, 2008.

[KKR08d] Ulrich Küster and Birgitta König-Ries. Towards standard test collections for the empirical evaluation of semantic web service approaches. *International Journal of Semantic Computing*, 2(3):381–402, September 2008.

[KKR09] Ulrich Küster and Birgitta König-Ries. Relevance judgments for web services retrieval — a methodology and test collection for sws discovery evaluation. In *Proceedings of the 7th IEEE European Conference on Web Services (ECOWS09)*, Eindhoven, The Netherlands, November 2009.

[KKR10] Ulrich Küster and Birgitta König-Ries. Measures for benchmarking semantic web service matchmaking correctness. In *Proceedings of the 7th Extended Semantic Web Conference (ESWC2010)*, Heraklion, Crete, Greece, May 2010.

[KKRK06a] Ulrich Küster, Birgitta König-Ries, and Michael Klein. Discovery and mediation using DIANE service descriptions. In *First Workshop of the Semantic Web Service Challenge 2006 - Challenge on Automating Web Services Mediation, Choreography and Discovery*, Palo Alto, California, USA, March 2006.

[KKRK06b] Ulrich Küster, Birgitta König-Ries, and Michael Klein. Discovery and mediation using DIANE service descriptions. In *Second Workshop of the Semantic Web Service Challenge 2006 - Challenge on Automating Web Services Mediation, Choreography and Discovery*, Budva, Montenegro, June 2006.

[KKRK08a] Ulrich Küster, Birgitta König-Ries, and Andreas Krug. OPOSSum — an online portal to collect and share semantic service descriptions. In *Proceedings of the 5th European Semantic Web Conference (ESWC08), Poster Session*, Tenerife, Canary Islands, Spain, June 2008.

[KKRK08b] Ulrich Küster, Birgitta König-Ries, and Andreas Krug. OPOSSum — an online portal to collect and share SWS descriptions. In *Proceedings of the 2nd IEEE International Conference on Semantic Computing (ICSC2008), Demo Session*, Santa Clara, CA, USA, August 2008.

[KKRK10] Ulrich Küster, Birgitta König-Ries, and Matthias Klusch. Evaluating semantic web service technologies: Criteria, approaches and challenges. In Miltiadis Lytras, editor, *Progressive Concepts for Semantic Web Evolution: Applications and Developments (Advances in Semantic Web Information Systems series)*. IGI Global, 2010.

[KKRKS07] Ulrich Küster, Birgitta König-Ries, Michael Klein, and Mirco Stern. DIANE — an integrated approach to automated service discovery, matchmaking and composition. In *Proceedings of the 16th International World Wide Web Conference (WWW2007)*, Banff, Alberta, Canada, May 2007.

[KKRKS08] Ulrich Küster, Birgitta König-Ries, Michael Klein, and Mirco Stern. DIANE — a matchmaking-centered framework for automated service discovery, composition, binding, and invocation on the web. *International Journal of Electronic Commerce (IJEC)*, 12(2):41–68, January 2008.

[KKRM05] Michael Klein, Birgitta König-Ries, and Michael Müssig. What is needed for semantic service descriptions — a proposal for suitable language constructs. *International Journal on Web and Grid Services (IJWGS)*, 1(3/4):328–364, 2005.

[KKRMS08] Ulrich Küster, Birgitta König-Ries, Tiziana Margaria, and Bernhard Steffen. Comparison: Handling preferences with DIANE and miAamics. In Charles Petrie, Holger Lausen, Michal Zaremba, and Tiziana

Margaria, editors, *Semantic Web Service Challenge — Results from the First Year*. Springer, 2008.

[KKRPK08] Ulrich Küster, Birgitta König-Ries, Charles Petrie, and Matthias Klusch. On the evaluation of semantic web service frameworks. *International Journal On Semantic Web and Information Systems*, 4(4):31–55, December 2008.

[KKRSK06] Ulrich Küster, Birgitta König-Ries, Mirco Stern, and Michael Klein. DIANE — a matchmaking-centered framework for automated service discovery, composition, binding, and invocation on the web. In *Proceedings of the 4th IEEE European Conference on Web Services (ECOWS06), Poster Session*, Zürich, Switzerland, December 2006.

[KKZ09] Matthias Klusch, Patrick Kapahnke, and Ingo Zinnikus. Hybrid adaptive web service selection with SAWSDL-MX and WSDL-Analyzer. In *Proceedings of the 6th European Semantic Web Conference (ESWC09)*, pages 550–564, Heraklion, Crete, Greece, June 2009.

[Kle06] Michael Klein. *Automatisierung dienstorientierten Rechnens durch semantische Dienstbeschreibungen (in German).* PhD thesis, Friedrich-Schiller-University Jena, Jena, Germany, 2006.

[KLKR07] Ulrich Küster, Holger Lausen, and Birgitta König-Ries. Evaluation of semantic service discovery - a survey and directions for future research. In *Proceedings of the 2nd Workshop on Emerging Web Services Technology (WEWST07) at the 5th IEEE European Conference on Web Services (ECOWS07)*, Halle (Saale), Germany, November 2007.

[KLL+05] Uwe Keller, Rubén Lara, Holger Lausen, Axel Polleres, and Dieter Fensel. Automatic location of services. In *Proceedings of the Second European Semantic Web Conference (ESWC2005)*, Heraklion, Crete, Greece, May 2005.

[Klu08a] Matthias Klusch. Semantic web service coordination. In H. Helin M. Schumacher, editor, *CASCOM - Intelligent Service Coordination in the Semantic Web*, chapter 4. Springer, 2008.

[Klu08b] Matthias Klusch. Semantic web service description. In H. Helin M. Schumacher, editor, *CASCOM - Intelligent Service Coordination in the Semantic Web*, chapter 3. Springer, 2008.

[KMK+08] Christian Kubczak, Tiziana Margaria, Matthias Kaiser, Jens Lemcke, and Björn Knuth. Abductive synthesis of the mediator scenario with jABC and GEM. In *Proceedings of the 6th International Workshop on Evaluation of Ontology-based Tools and the Semantic Web Service Challenge (EON-SWSC08)*, Tenerife, Canary Islands, Spain, June 2008.

[KMS+08] Christian Kubczak, Tiziana Margaria, Bernhard Steffen, Christian Winkler, and Hardi Hungar. An approach to discovery with miAamics and jABC. In Charles Petrie, Holger Lausen, Michal Zaremba, and Tiziana Margaria, editors, *Semantic Web Service Challenge — Results from the First Year*. Springer, 2008.

[Kol08] Dave Kolas. A benchmark for spatial semantic web systems. In *Proceedings of the 4th International Workshop on Scalable Semantic Web Knowledge Base Systems (SSWS2008)*, October 2008.

[KSKR05] Ulrich Küster, Mirco Stern, and Birgitta König-Ries. A classification of issues and approaches in service composition. In *Proceedings of the First International Workshop on Engineering Service Compositions (WESC05)*, Amsterdam, Netherlands, December 2005.

[KTKR+08] Ulrich Küster, Andrea Turati, Birgitta König-Ries, Dario Cerizza, Emanuele Della Valle, and Federico M. Facca. Service discovery with SWE-ET and DIANE — an in-depth comparison by means of a common scenario. In Charles Petrie, Holger Lausen, Michal Zaremba, and Tiziana Margaria, editors, *Semantic Web Service Challenge — Results from the First Year*. Springer, 2008.

[KTSW08] Dominik Kuropka, Peter Tröger, Steffen Staab, and Mathias Weske, editors. *Semantic Service Provisioning*. Springer, 2008.

[KTZ+07] Ulrich Küster, Andrea Turati, Maciej Zaremba, Birgitta König-Ries, Dario Cerizza, Emanuele Della Valle, Marco Brambilla, Stefano Ceri, Federico Facca, and Christina Tziviskou. Service discovery with SWE-ET and DIANE — a comparative evaluation by means of solutions to a common scenario. In *Proceedings of the 9th International Conference on Enterprise Information Systems (ICEIS2007), Special Session on Comparative Evaluation of Semantic Web Service Frameworks*, Funchal, Madeira-Portugal, June 2007.

[KvdHD06a] Natallia Kokash, Willem-Jan van den Heuvel, and Vincenzo D'Andrea. Leveraging web services discovery with customizable hybrid matching. In *Proceedings of the 4th International Conference*

on Service-Oriented Computing (ICSOC06), Short Paper, pages 522–528, Chicago, IL, USA, December 2006.

[KvdHD06b] Natallia Kokash, Willem-Jan van den Heuvel, and Vincenzo D'Andrea. Leveraging web services discovery with customizable hybrid matching. Technical Report DIT-06-042, DIT-University of Trento, Trento, Italy, July 2006.

[KVW+08] Christian Kubczak, Tomas Vitvar, Christian Winkler, Raluca Zaharia, and Maciej Zaremba. Comparison: Discovery on WSMOLX and miAamics/jABC. In Charles Petrie, Holger Lausen, Michal Zaremba, and Tiziana Margaria, editors, *Semantic Web Service Challenge — Results from the First Year*. Springer, 2008.

[KZ08] Matthias Klusch and Xiguo Zhing. Deployed semantic services for the common user of the web: A reality check. In *Proceedings of the 2nd IEEE International Conference on Semantic Computing (ICSC2008)*, Santa Clara, CA, USA, August 2008.

[Lar06] Ruben Lara. Two-phased web service discovery. In *Proceedings of the workshop on AI-Driven Technologies for Service-Oriented Computing at the Twenty First National Conference on Artificial Intelligence (AAAI06)*, Boston, USA, July 2006.

[LBC+05] Ruben Lara, Walter Binder, Ion Constantinescu, Dieter Fensel, Uwe Keller, Jeff Pan, Marco Pistore, Axel Polleres, Ioan Toma, Paolo Traverso, and Michal Zaremba. Knowledge Web Deliverable D2.4.2: Semantics for web service discovery and composition. Technical Report KWEB/2005/D2.4.2/v1.1, KWEB EU-IST-2004-507482, January 2005.

[LCMdF+08] José-Manuell López-Cobo, Iván Martínez, David de Francisco, Bernhard Schreder, and SinueArroyo. SUPER Deliverable 8.4: YATOSP validation plan / technical report. Technical report, Project IST 026850 SUPER, October 2008.

[LH03] Lei Li and Ian Horrocks. A software framework for matchmaking based on semantic web technology. In *Proceedings of the 12th World Wide Web Conference (WWW2003)*, Budapest, Hungary, May 2003.

[LKF06] Holger Lausen, Uwe Keller, and Dieter Fensel. RW^2 Project Deliverable D2.2 v1.0: Discovery framework specification. Technical report, FIT-IT Programme Project RW^2, July 2006.

[LKP+08] Holger Lausen, Ulrich Küster, Charles Petrie, Michal Zaremba, and Srdjan Komazec. Sws challenge scenarios. In Charles Petrie, Holger Lausen, Michal Zaremba, and Tiziana Margaria, editors, *Semantic Web Service Challenge — Results from the First Year*. Springer, 2008.

[LPZ07] Holger Lausen, Charles Petrie, and Michal Zaremba. W3C SWS testbed incubator group charter. available online at http://www.w3.org/2005/Incubator/swsc/charter, 2007.

[Lut02] Carsten Lutz. Description logics with concrete domains — a survey. In *Advances in Modal Logic 4, Papers from the Fourth Conference on Advances in Modal Logic*, pages 265–296, Toulouse, France, October 2002.

[MDGM06] Vincenzo Della Mea, Gianluca Demartini, Luca Di Gaspero, and Stefano Mizzaro. Measuring retrieval effectiveness with average distance measure (ADM). *Information Wissenschaft und Praxis*, 57(8):405–416, 2006.

[Mel07] Massimo Melucci. On rank correlation in information retrieval evaluation. *ACM SIGIR Forum*, 41(1):18–33, 2007.

[Mik07] Peter Mika. *Social Networks and the Semantic Web*. Springer, 2007.

[Miz97] Stefano Mizzaro. Relevance: The whole history. *JASIS*, 48(9):810–832, 1997.

[MM07] Ioana Manolescu and Stefan Manegold. Performance evaluation and experimental assessment - conscience or curse of database research? In *VLDB*, pages 1441–1442, 2007.

[MSCV06] Adrian Mocan, Francois Scharffe, Emilia Cimpian, and Tomas Vitvar. Knowledge Web Deliverable D2.4.12: Data mediation in semantic web services. Technical Report KWEB/2006/D2.4.12/v1.0, KWEB EU-IST-2004-507482, December 2006.

[MSJL06] James McGovern, Oliver Sims, Ashish Jain, and Mark Little. *Enterprise Service Oriented Architectures - Concepts, Challenges, Recommendations*. Springer, 2006.

[MSZ01] Sheila A. McIlraith, Tran Cao Son, and Honglei Zeng. Semantic web services. *IEEE Intelligent Systems*, 16(2):46–53, 2001.

[MYQ+06] Li Ma, Yang Yang, Zhaoming Qiu, Guo Tong Xie, Yue Pan, and Shengping Liu. Towards a complete OWL ontology benchmark. In *Proceedings of the 3rd European Semantic Web Conference (ESWC2006)*, pages 125–139, Budva, Montenegro, June 2006.

[NS76] Allen Newell and Herbert A. Simon. Computer science as empirical inquiry: symbols and search. *Communications of the ACM*, 19(3):113–126, 1976.

[NSD07] Tommaso Di Noia, Eugenio Di Sciascio, and Francesco M. Donini. Semantic matchmaking as non-monotonic reasoning: A description logic approach. *Journal of Artificial Intelligence Research (JAIR)*, 29:269–307, 2007.

[NSDM03] Tommaso Di Noia, Eugenio Di Sciascio, Francesco M. Donini, and Marina Mongiello. A system for principled matchmaking in an electronic marketplace. In *Proceedings of the Twelfth International World Wide Web Conference (WWW2003)*, Budapest, Hungary, May 2003.

[OLES05] Daniel Oberle, Steffen Lamparter, Andreas Eberhart, and Steffen Staab. Semantic management of web services. In *Proceedings of the 2005 IEEE International Conference on Web Services (ICWS05)*, Orlando, Florida, USA, July 2005.

[Pap08] Mike P. Papazoglou. *Web Services: Principles And Technology*. Pearson Education, 2008.

[PCB+05] Chris Preist, Javier Esplugas Cuadrado, Steven A. Battle, Stephan Grimm, and Stuart K. Williams. Automated business-to-business integration of a logistics supply chain using semantic web services technology. In *Proceedings of the Fourth International Semantic Web Conference*, Galway, Ireland, November 2005.

[Pee05] Joachim Peer. Web service composition as AI planning — a survey. Technical report, University of St. Gallen, Switzerland, 2005.

[Pet06] Charles Petrie. It's the programming, stupid. *IEEE Internet Computing*, 10(3):95–96, 2006.

[Pet08] Charles Petrie. Introduction to the first year of the semantic web services challenge. In Charles Petrie, Holger Lausen, Michal Zaremba, and Tiziana Margaria, editors, *Semantic Web Service Challenge — Results from the First Year*. Springer, 2008.

[PGF96] Robert E. Park, Wolfhart B. Goethert, and William A. Florac. Goal-driven software measurement — a guidebook. Technical report, Carnegie Mellon University, Pittsburgh, PA, USA, August 1996.

[PKM+08] Charles Petrie, Ulrich Küster, Tiziana Margaria, Michal Zaremba, Holger Lausen, and Srdjan Komazec. Status, perspectives, and lessons learned. In Charles Petrie, Holger Lausen, Michal Zaremba, and Tiziana Margaria, editors, *Semantic Web Service Challenge — Results from the First Year*. Springer, 2008.

[PKMS08] Charles Petrie, Ulrich Küster, and Tiziana Margaria-Steffen. W3C SWS challenge testbed incubator methodology report. W3c incubator report, W3C, March 2008. available online at http://www.w3.org/2005/Incubator/swsc/XGR-SWSC/.

[PKPS02] Massimo Paolucci, Takahiro Kawamura, Terry R. Payne, and Katia P. Sycara. Semantic matching of web services capabilities. In *Proceedings of the First International Semantic Web Conference (ISWC2002)*, pages 333–347, Sardinia, Italy, June 2002.

[PLZM08] Charles Petrie, Holger Lausen, Michal Zaremba, and Tiziana Margaria, editors. *Semantic Web Service Challenge — Results from the First Year*. Springer, Semantic Web and Beyond, Vol. 8, 2008.

[PMK+07] Charles Petrie, Tiziana Margaria, Ulrich Küster, Holger Lausen, and Michal Zaremba. SWS Challenge: status, perspectives and lessons learned so far. In *Proceedings of the 9th International Conference on Enterprise Information Systems (ICEIS2007), Special Session on Comparative Evaluation of Semantic Web Service Frameworks*, Funchal, Madeira-Portugal, June 2007.

[Pre04] Chris Preist. A conceptual architecture for semantic web services (extended version). Technical Report HPL-2004-215, HP Laboratories Bristol, November 2004.

[Pre07] Chris Preist. Goals and vision: Combining web services with semantic web technology. In *Semantic Web Services: Concepts, Technologies, and Applications*, pages 159–178. Springer, 2007.

[PSK03] Massimo Paolucci, Katia P. Sycara, and Takahiro Kawamura. Delivering semantic web services. In *Proceedings of the Twelfth International World Wide Web Conference (Alternate Paper Tracks)*, Budapest, Hungary, May 2003.

[Ric07] Marc Richardson. WP 8: B2B in telecommunications, D8.6b: Contract catalogue prototype v2. Technical report, DIP Project FP6 - 507483, January 2007.

[RSN+07] Azzurra Ragone, Umberto Straccia, Tommaso Di Noia, Eugenio Di Sciascio, and Francesco M. Donini. Vague knowledge bases for matchmaking in P2P e-marketplaces. In *Proceedings of the 4th European Semantic Web Conference (ESWC2007)*, Innsbruck, Austria, June 2007.

[SAG07] Rudi Studer, Andreas Abecker, and Stephan Grimm, editors. *Semantic Web Services — Concepts, Technologies and Applications*. Springer, 2007.

[Sak04a] Tetsuya Sakai. New performance metrics based on multigrade relevance: Their application to question answering. In *Proceedings of the Fourth NTCIR Workshop on Research in Information Access Technologies, Information Retrieval, Question Answering and Summarization (NTCIR04)*, Tokyo, Japan, June 2004.

[Sak04b] Tetsuya Sakai. Ranking the NTCIR systems based on multigrade relevance. In *Revised Selected Papers of the Asia Information Retrieval Symposium*, pages 251–262, Beijing, China, October 2004.

[Sak07a] Tesuya Sakai. On penalising late arrival of relevant documents in information retrieval evaluation with graded relevance. In *Proceedings of the First International Workshop on Evaluating Information Access (EVIA)*, Tokyo, Japan, May 2007.

[Sak07b] Tetsuya Sakai. On the reliability of information retrieval metrics based on graded relevance. *Information Processing and Management*, 43(2):531–548, 2007.

[Sar95] Tefko Saracevic. Evaluation of evaluation in information retrieval. In *Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR95)*, pages 138–146, Seattle, Washington, USA, July 1995.

[Sar07a] Tefko Saracevic. Relevance: A review of the literature and a framework for thinking on the notion in information science. part ii: nature and manifestations of relevance. *Journal of the American Society for Information Science and Technology*, 58(13):1915–1933, 2007.

[Sar07b] Tefko Saracevic. Relevance: A review of the literature and a framework for thinking on the notion in information science. part iii: Behavior and effects of relevance. *Journal of the American Society for Information Science and Technology*, 58(13):2126–2144, 2007.

[Sar08] Tefko Saracevic. Effects of inconsistent relevance judgments on information retrieval test results: A historical perspective. *Library Trends*, 56(4):763–783, 2008.

[SC07] Zulima Saenz and Cyril Carrez. SIMS Deliverable 6.1: Evaluation and assessment plan. Technical report, FP6 - 027610 SIMS, Febuary 2007.

[SEH03] Susan Elliott Sim, Steve M. Easterbrook, and Richard C. Holt. Using benchmarking to advance research: A challenge to software engineering. In *Proceedings of the 25th International Conference on Software Engineering (ICSE2003)*, pages 74–83, Portland, Oregon, USA, May 2003.

[Ser93] Omri Serlin. *The History of DebitCredit and the TPC*. Morgan Kaufmann, 1993.

[Shi08a] Mazen Malek Shiaa. SIMS Deliverable 6.5: Trial services, final version. Technical report, FP6 - 027610 SIMS, November 2008.

[Shi08b] Mazen Malek Shiaa. SIMS Deliverable 6.6: Evaluation of SIMS approach. Technical report, FP6 - 027610 SIMS, November 2008.

[SHLP09] Michael Schmidt, Thomas Hornung, Georg Lausen, and Christoph Pinkel. SP^{2}Bench: A SPARQL performance benchmark. In *Proceedings of the 25th International Conference on Data Engineering (ICDE2009)*, pages 222–233, Shanghai, China, March 2009.

[SIG07] SIGMOD. SIGMOD 2008 experimental repeatability requirements. *SIGMOD Record*, 36(2), 2007.

[Sim03] Susan Elliott Sim. *A Theory of Benchmarking with Applications to Software Reverse Engineering*. PhD thesis, Department of Computer Science, University of Toronto, 2003.

[SJ95] Karen Sparck Jones. Reflections on TREC. *Information Processing and Management*, 31(3):291–314, 1995.

[SKFL07] Nathalie Steinmetz, Uwe Keller, Cristina Feier, and Holger Lausen. RW^2 Project Deliverable D1.4 v1.0: Evaluation of the reasoning procedures and techniques. Technical report, FIT-IT Programme Project RW^2, July 2007.

[SNSW94] Mary Ann Scheirer, Dianna Newman, William Shadish, and Chris Wye. Guiding principles for evaluators. Technical report, American Evaluation Association, 1994.

[SPvS09] Eduardo Silva, Luis Ferreira Pires, and Marten van Sinderen. A framework for the evaluation of semantics-based service composition approaches. In *Proceedings of the 7th IEEE European Conference on Web Services (ECOWS09)*, Eindhoven, The Netherlands, November 2009.

[SR08] Tetsuya Sakai and Stephen Robertson. Modelling a user population for designing information retrieval metrics. In *Proceedings of the Second International Workshop on Evaluating Information Acess (EVIA08)*, Tokyo, Japan, December 2008.

[SS06] Natenapa Sriharee and Twittie Senivongse. Matchmaking and ranking of semantic web services using integrated service profile. *International Journal on Metadata, Semantics and Ontologies*, 1(2):100–118, 2006.

[STK$^+$04] Michael Stollberg, Ioan Toma, Uwe Keller, Bernhard Keimel, and Peter Zugmann. D3.5 SWF use case — final version 1.2. available at http://swf.deri.at/usecase/20041019/, October 2004.

[STR06] Adina Sîrbu, Ioan Toma, and Dumitru Roman. A logic based approach for service discovery with composition support. In *Proceedings of the ECOWS06 Workshop on Emerging Web Services Technology*, Zürich, Switzerland, December 2006.

[SW05] Eleni Stroulia and Yiqiao Wang. Structural and semantic matching for assessing web-service similarity. *International Journal of Cooperative Information Systems (IJCIS)*, 14(4):407–438, 2005.

[TAH06] Vassileios Tsetsos, Christos Anagnostopoulos, and Stathes Hadjiefthymiades. On the evaluation of semantic web service matchmaking systems. In *Proceedings of the 4th IEEE European Conference on Web Services (ECOWS2006)*, Zürich, Switzerland, December 2006.

[Tic98] Walter F. Tichy. Should computer scientists experiment more? *IEEE Computer*, 31(5):32–40, May 1998.

[Tid00] D. Tidwell. Web services: The web's next revolution. Technical report, IBM developerworks, 2000.

[TIR+07] Ioan Toma, Kashif Iqbal, Dumitru Roman, Thomas Strang, Dieter Fensel, Brahmananda Sapkota, Matthew Moran, and Juan Miguel Gomez. Discovery in grid and web services environments: A survey and evaluation. *International Journal on Multiagent and Grid Systems*, 3(3), 2007.

[TIT+05] Ioan Toma, Kashif Iqbal, Bernhard Tausch, Jarno Heikkilä, and Dumitru Roman. Adaptive Services Grid Deliverable D1.I-2: Evaluation of current effort in service and resource matchmaking. Technical report, Adaptive Services Grid Project FP6 - 004617, February 2005.

[TLPH95] Walter F. Tichy, Paul Lukowicz, Lutz Prechelt, and Ernst A. Heinz. Experimental evaluation in computer science: a quantitative study. *Journal of Systems and Software*, 28(1):9–18, 1995.

[TRF06] Ioan Toma, Dumitru Roman, and Dieter Fensel. Modeling semantic web services in ASG: The WSMO-based approach. In S. Reich, G. Guentner, T. Pellegrini, and A. Wahler, editors, *Semantic Content Engineering - Proceedings of Semantics2005*, Linz, 2006. Trauner Verlag.

[TS92] Jean Tague-Sutcliffe. The pragmatics of information retrieval experimentation, revisited. *Information Processing and Management*, 28(4):467–490, 1992.

[TVCF08] Andrea Turati, Emanuele Della Valle, Dario Cerizza, and Federico M. Facca. Using glue to solve the discovery scenarios of the SWS-Challenge. In Charles Petrie, Holger Lausen, Michal Zaremba, and Tiziana Margaria, editors, *Semantic Web Service Challenge — Results from the First Year*. Springer, 2008.

[UB09] M. Urvois and Arne J. Berre. SWING Deliverable 1.3: Experience report. Technical report, FP6 - 26514 SWING, July 2009.

[Uns06] Unspecified. WP 10: Case study ebanking, D10.10: Evaluation of application 2. Technical report, DIP Project FP6 - 507483, December 2006.

[VHA05] Le-Hung Vu, Manfred Hauswirth, and Karl Aberer. QoS-based service selection and ranking with trust and reputation management. In *Proceedings of the OTM Confederated International Conferences CoopIS,*

DOA, and ODBASE 2005, Agia Napa, Cyprus, October, November 2005.

[VLZP06] Tomas Vitvar, Holger Lausen, Michal Zaremba, and Charles Petrie. Knowledge Web Deliverable D2.4.13: Report on the semantic web services challenge. Technical Report KWEB/2006/D2.4.13/v1.0, KWEB EU-IST-2004-507482, June 2006.

[VMZ+07] Tomas Vitvar, Matthew Moran, Maciej Zaremba, Michal Zaremba, Adrian Mocan, Mick Kerrigan, and Thomas Hasselwanter. Knowledge Web Deliverable D2.4.10: Architecture and execution semantics for the SWS. Technical Report KWEB/2006/D2.4.10/v2, KWEB EU-IST-2004-507482, December 2007.

[Voo98] Ellen M. Voorhees. Variations in relevance judgments and the measurement of retrieval effectiveness. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR98)*, pages 315–323, Melbourne, Australia, 1998.

[Voo01a] Ellen M. Voorhees. Evaluation by highly relevant documents. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR01)*, pages 74–82, New Orleans, LA, USA, September 2001.

[Voo01b] Ellen M. Voorhees. The philosophy of information retrieval evaluation. In *Evaluation of Cross-Language Information Retrieval Systems, Second Workshop of the Cross-Language Evaluation Forum (CLEF 2001)*, pages 355–370, Darmstadt, Germany, September 2001.

[Voo05] Ellen M. Voorhees. TREC: improving information access through evaluation. *Bulletin of the American Society for Information Science and Technology*, 32, October/November 2005. 1.

[VS07] Denny Vrandecic and York Sure. How to design better ontology metrics. In *Proceedings of the 4th European Semantic Web Conference (ESWC2007)*, pages 311–325, Innsbruck, Austria, June 2007.

[VTe06] Jari Veijalainen Ville Törmälä and Holger Krause (editors). Adaptive Services Grid Deliverable D7.IV: Business analysis, deployment strategy and evaluation of ASG based services – part a: Scenarios and business. Technical report, Adaptive Services Grid Project FP6 - 004617, September 2006.

[VVSH08] Johanna Völker, Denny Vrandecic, York Sure, and Andreas Hotho. AEON an approach to the automatic evaluation of ontologies. *Journal of Applied Ontology*, 3(1–2):41–62, 2008.

[WAP+02] Claes Wohlin, Aybuke Aurum, Håkan Petersson, Forrest Shull, and Marcus Ciolkowski. Software inspection benchmarking — a qualitative and quantitative comparative opportunity. In *Proceedings of the 8th International Symposium on Software Metrics (Metrics02)*, Ottawa, Canada, June 2002.

[War96] Steve Wartik. Are comparative analyses worthwhile? *IEEE Computer*, 29(7):120, 1996.

[WDR+04] Sam Watkins, Alistair Duke, Marc Richardson, Bernhard Schreder, Alexander Wahler, Ruben Verlinden, and Thomas Haselwanter. WP 8: Case study B2B in telecommunications, D8.4: Case study implementation prototype v1. Technical report, DIP Project FP6- 507483, July 2004.

[WHBK87] Nelson H. Weiderman, A. Nico Habermann, Mark W. Borger, and Mark H. Klein. A methodology for evaluating environments. In *SDE 2: Proceedings of the second ACM SIGSOFT/SIGPLAN software engineering symposium on Practical software development environments*, pages 199–207, New York, NY, USA, 1987.

[WRH+00] Claes Wohlin, Per Runeson, Martin Höst, Magnus C. Ohlsson, Björn Regnell, and Anders Wesslén. *Experimentation in Software Engineering - An Introduction.* Kluver Academic Publishers, 2000.

[Zel06] Marvin V. Zelkowitz. Techniques for empirical validation. In *Revised Papers of the International Dagstuhl Workshop on Empirical Software Engineering Issues. Critical Assessment and Future Directions*, pages 4–9. Springer, Dagstuhl Castle, Germany, June 2006.

[Zel08] Marvin V. Zelkowitz. An update to experimental models for validating computer technology. *The Journal of Systems and Software*, 2008.

[ZMV08] Maciej Zaremba, Matthew Moran, and Tomas Vitvar. Instance-based service discovery with WSMO/WSML and WSMX. In Charles Petrie, Holger Lausen, Michal Zaremba, and Tiziana Margaria, editors, *Semantic Web Service Challenge — Results from the First Year.* Springer, 2008.

[ZW97] Marvin V. Zelkowitz and Dolores R. Wallace. Experimental validation in software engineering. *Information & Software Technology*, 39(11):735–743, 1997.

[ZW98a] Marvin V. Zelkowitz and Dolores Wallace. Validating the benefit of new software technology. *Software Quality Practitioner*, 1(1), 1998.

[ZW98b] Marvin V. Zelkowitz and Dolores R. Wallace. Experimental models for validating technology. *Computer*, 31(5):23–31, 1998.

Appendix

APPENDIX A

Analysis of SWS Evaluation Campaigns by Evaluation Requirements

This appendix provides an extended analysis of the SWS Challenge, the S3 Contest and the WS Challenge initiatives with respect to the evaluation requirements identified in Section 4.2. For each of the initiatives, the requirements will be discussed by briefly stating the requirement and providing a short statement with respect to the questions operationalizing the requirement. For improved legibility, the questions are not repeated but are available in Section 4.2. The analysis presented here served as the basis for the corresponding summary in Section 4.4.

A.1. SWS Challenge

Please note that the analysis of the SWS Challenge represents the state as of end of 2009. Since the discovery track of the SWS Challenge was partially developed as part of this thesis work we will focus on the complementary mediation track. The discovery track will be analyzed with respect to the requirements as part of the validation of this thesis in Section 8.6.

A.1.1. Utility Requirements

The Utility Requirements are intended to ensure that an evaluation is guided by both the clarified purposes of the evaluation and the information needs of its intended users.

Utility 1 (Stakeholder Identification) *Persons or groups potentially interested in or affected by the evaluation should be identified and contacted, so that their interests can be clarified and taken into consideration when designing the evaluation.*

The scope of the evaluation, potential participants and users of the evaluation results are defined. People in the community were repeatedly contacted by email, in person at conferences and through common mailing lists in the community. Some probe contacts with industrial players in the field (e.g., SAP) have not indicated high interest and have thus not been further pursued. There were several calls for general participation and involvement in the benchmark design was encouraged and possible through associated mailing lists and discussion at workshops.

Utility 2 (Clarification of the Purposes of the Evaluation) *The purposes of the evaluation shall be stated clearly, so that the stakeholders can provide relevant comments on these purposes, and so that the evaluation team knows exactly what it is expected to do.*

The goals and purposes of the evaluations are clearly defined and the usage context described. Evaluation results provide insights about the technology characteristics and development practices that lead to them via the papers describing the solutions to the benchmark problems and especially via the additional in-depth comparisons of solutions that have been prepared. Results do not provide explicit technology improvement recommendations because such interpretation is left to the participants. However, improvement of the evaluated tools or techniques is an explicit goal of the evaluation.

Utility 3 (Evaluator Credibility and Competence) *The persons conducting an evaluation shall be trustworthy as well as methodologically and professionally competent, so that the evaluation findings achieve maximum credibility and acceptance.*

According to the benchmark methodology, the evaluation team comprises the whole set of participants at each evaluation workshop. The general organizing team of the initiative comprises roughly fifteen people from seven institutions and thus reflects the diversity of the interested research community. The organization team is open for new people anytime and in fact has changed over time.

Utility 4 (Information Scope and Selection) *The scope and selection of the collected information shall make it possible to answer relevant questions about the evaluand and consider the information needs of the client and other stakeholders.*

The design of the evaluation was discussed at several open workshops and also within a W3C Incubator activity. To the best of our knowledge, there are no directly comparable evaluation approaches. The problem scenarios defined by the benchmark are believed to be representative problems for the expected usage of SWS in industry. However, the selection of scenarios is supported neither by empirical work nor a model nor theory. Unlike with the scenarios of the discovery track, the mediation scenarios have not been approved by a workshop. Furthermore, the scenarios are not divided into subproblems. Participants can choose among two scenario branches, but apart from that, the evaluation can not be scaled to be more or less complex. The tasks can be solved even using traditional programming but an exemplary solution is not provided as part of the evaluation setup. Good and poor solutions with respect to the necessary programming effort and the adaptability of the resulting program to change are possible.

Utility 5 (Transparency of Values) *The perspectives and assumptions of the stakeholders that serve as a basis for the evaluation and the interpretation of the evaluation findings shall be described in a way that clarifies their underlying values.*

The assumptions behind the evaluation use case scenarios are described in the problem scenario definitions. The design goals of the evaluated technologies are described in detail in the papers accompanying every benchmark entry. Furthermore, the detailed comparison of solutions includes a comparison of these design goals.

Utility 6 (Report Comprehensiveness and Clarity) *Evaluation reports shall provide all relevant information and be easily comprehensible.*

The setup of the evaluation regarding prerequisites, assumptions, input data, roles and tasks is described. The process of the information collection, i.e., the actual certification procedure, is described in detail in a W3C Incubator Group report [PKMS08]. However, changes made over time in the evaluation setup have not always been reflected in the documentation of the scenarios. Furthermore, information is somewhat scattered over various web sites and the mentioned document. It has to be suspected that the available information is insufficient to make the evaluation completely comprehensible for interested outsiders.

Utility 7 (Evaluation Timeliness) *The evaluation shall be initiated and completed in a timely fashion so that its findings can inform pending decision and improvement processes.*

The scenarios are open for solutions any time and evaluation workshops have been held at least yearly since 2006. Scheduling of subsequent workshops is planned during workshops in a way that tries to accommodate the preferences of everybody involved and potential new participants. There are discussion slots at every workshop where feedback about the evaluation is collected. The benchmark has been updated several times.

Utility 8 (Evaluation Utilization and Use) *The evaluation shall be planned, conducted, and reported in ways that encourage attentive follow-through by stakeholders and utilization of the evaluation findings.*

The SWS Challenge was publicized on mailing lists, at conferences and through private communication. It has been held in conjunction with several major conferences in the area. It provides a wiki with comprehensive information and several mailing lists to accommodate different levels of involvement in the communication about the initiative. Participation in the benchmarking is open to everyone. The evaluation can be executed offline only in a limited way, however, evaluation events are repeated at least on a yearly basis. There is no fee or other restriction for participation, but attendance of a workshop is required. Depending on the venue this has sometimes required a workshop or conference registration fee in addition to the other travel expenses.

Results and findings of the evaluation are discussed at open workshops and published on the wiki. All available information is freely accessible. However, the testbed supporting the evaluation is not very well documented and its source code not publicly available. Effectively, without the support from the group hosting the testbed (currently STI Innsbruck, Austria), the evaluation can be executed only very limitedly. Furthermore, solutions to the scenarios have not always been documented properly. This limits the extent to which the evaluation results can be comprehended and reproduced.

A.1.2. Feasibility Requirements

The Feasibility Requirements are intended to ensure than an evaluation is planned and conducted in a realistic, thoughtful, diplomatic and cost-effective manner.

Feasibility 1 (Appropriate Procedures) *Evaluation procedures, including information collection procedures, shall be chosen so that the burden and cost placed on the stakeholders is appropriate in comparison to the expected benefits of the evaluation.*

A discussion of the pros and cons of the evaluation setup is available in [PKMS08] but the relevance of the chosen procedure and the chosen scenarios is not discussed extensively, in particular with respect to the evaluation goal of determining the flexibility of solutions and the advantage of semantics over traditional programming. A testbed supporting the evaluation is available, but an automated correctness check of solutions has not been implemented yet.

Feasibility 2 (Diplomatic Conduct) *The evaluation shall be planned and conducted so that it achieves maximal acceptance by the different stakeholders with regard to the evaluation process and findings.*

The setup of the evaluation campaign explicitly encourages participants to reuse each other's results, to form teams and to learn about each other's approaches by preparing jointly authored comparisons of the various technologies. The initiative is organized as a challenge rather than a contest and there is no declared winner of the evaluation event. Presentation of evaluation results is discussed at each workshop before the certification results are put online.

Feasibility 3 (Evaluation Efficiency) *The relation between cost and benefit of the evaluation shall be appropriate.*

Participants in the evaluation need skills about SOAP based Web services and to some extent about RosettaNet protocol standards. Otherwise, no skills except for their own technologies are required. Unfortunately, no estimates of the minimal or optimal predicted time necessary to solve a problem scenarios or other explicit cost-benefit have been provided so far. Monetary cost is involved for the mandatory participation in an evaluation workshop. The problem scenarios are only to a very limited extent organized in a way that allows scaling the evaluation to be more or less complex.

A.1.3. Propriety Requirements

The Propriety Requirements are intended to ensure that in the course of the evaluation all stakeholders are treated with respect and fairness, that the evaluation achieves maximum objectivity and provides an unbiased and appropriate analysis of the technologies under examination.

Propriety 1 (Formal Agreement) *Obligations of the formal parties to an evaluation shall be agreed to in writing, so that these parties are obligated to adhere to all conditions of the agreement or to renegotiate it.*

No written agreement about the responsibilities and commitments of everyone involved in the evaluation has been prepared and the responsibilities and commitments of everyone have not always been clear.

Propriety 2 (Protection of Individual Rights) *The evaluation shall be designed and conducted in a way that protects the welfare, dignity and rights of all stakeholders.*

Results are discussed at the workshops before they are released. Furthermore, participants are encouraged to vet their results. There is no explicit process or option for users to veto the release of their results. However, so far, this has never been an issue.

Propriety 3 (Complete and Fair Investigation) *The evaluation shall undertake a complete and fair examination and description of strengths and weaknesses of the evaluand so that strengths can be built upon and problem areas addressed.*

The evaluation results provide detailed information about the tools under investigation. Since participants are required to prepare a paper about their entry, they have the option to give appropriate coverage of the strengths and weaknesses of the evaluated technology. Detailed and meaningful comparisons of the evaluated technologies are furthermore provided by means of jointly written comparison papers. However, the selection of characteristics of a technology under evaluation is determined by the characteristics of the currently available problem scenarios and not further justified.

Propriety 4 (Unbiased Conduct and Reporting) *The evaluation shall take into account the different views of the stakeholders concerning the evaluand and the evaluation findings. Like the entire evaluation process, the evaluation report shall evidence the impartial position of the evaluation team. Value judgments shall be made as unemotionally as possible.*

The evaluation is largely independent from any particular solution. None of the scenarios has been developed by teams also participating in the evaluation. The evaluation is entirely independent from a particular platform or technology except that the scenarios require the interaction with standard SOAP based Web services. Evaluation scenarios are specified exclusively using natural language English text and standard XML schemata and WSDL files. They are thus applicable to all technologies and not biased in favor of a specific one. A reviewing process for evaluation results and reports is in place (results are discussed at the workshops

and participants are encouraged to review the published information). However, with respect to the adaptability of solutions and the effort required to react to changes in the problem scenarios, the evaluation is difficult to apply to research prototypes in the same way as to mature products. Also the evaluation does not really provide tasks or inputs at different complexity or size.

Propriety 5 (Disclosure of Findings) *As far as possible, all stakeholders shall have access to the evaluation findings.*

Scores and evaluation results are publicly documented. Participants are encouraged to upload their solutions to an FTP server and document it in a way that allows the independent reproducibility of the evaluation findings. However, this process is not mandatory and most solutions have not been documented sufficiently to allow for independent reproducibility of evaluation results. The terms of publication of evaluation results are defined in the calls for participation and the general description of the benchmark. Publication of results has been performed accordingly. All information collected during the benchmarking is publicly available.

A.1.4. Accuracy Requirements

The accuracy requirements are intended to ensure that an evaluation produces and discloses valid, accurate, precise, reliable and useful information and findings pertaining to the evaluation purposes and questions.

Accuracy 1 (Description of the Evaluand) *The evaluand shall be described and documented clearly and accurately so that it can be unequivocally identified.*

The benchmark clearly identifies the tasks to perform and the characteristics of a technology under investigation. The quality criteria being assessed in the evaluation are clearly defined and the tools or techniques that are intended to be evaluated specified. The evaluation requirements and assumptions are provided.

Accuracy 2 (Context Analysis) *The context of the technologies being evaluated shall be examined and analyzed in sufficient detail.*

The context of the technologies being evaluated is not examined by the initiative organizers as part of the evaluation. However, participants are encouraged to discuss these and their potential influence on the evaluation findings in the paper and solution documentation accompanying any entry.

Accuracy 3 (Described Purposes and Procedures) *Object, purposes, methodology and procedures of an evaluation, including the applied methods, shall be accurately documented and described so that they can be identified and assessed.*

The measured characteristics and evaluation criteria as well as the tasks to perform and the input data are clearly defined in the benchmark specification. However, the means and measures used to assess the flexibility and adaptability of solutions are not discussed in sufficient detail. It is not entirely clear how the corresponding evaluation results are to be interpreted. Furthermore, the testbed supporting the evaluation is not very well documented and described.

Accuracy 4 (Disclosure of Information Sources) *The information sources used in the course of the evaluation shall be documented in appropriate detail so that the reliability and adequacy of the information can be assessed.*

The raw data on which the evaluation findings are based consist primarily of the code review and demonstration performed during the workshop. This "raw data" is so far not documented (e.g., filmed or otherwise protocolled) in detail. Thus, while it is traceable how the raw data was obtained, it is not easy to verify the quality of the process and data retrospectively. Furthermore, the server logs that protocol the correct communication of solutions with the testbed are only partially available.

Accuracy 5 (Valid and Reliable Information) *The data collection procedures shall be chosen and developed and then applied in a way that ensures the reliability and validity of the data with regard to answering the evaluation questions. The technical criteria shall be based on the standards of quantitative and qualitative social research.*

The assumptions made by the evaluation are deemed to be realistic even though the scenarios have not been approved by a community process as in the discovery track. However, the selection of performance measures for measuring the flexibility and adaptability of solutions have been justified only partially. While one person applying the evaluation on the same technology would very likely get the same results twice, different people applying the evaluation on the same technology may get different results depending on how clever they use the technology when designing their solution architecture. Some solutions have been criticized for awkward usage of the employed technology. The corresponding threats to the evaluation's validity are not discussed in depth. With respect to adaptability, it is also not clear to what extent the results are affected by the quality of tool support and maturity of implementation. Supposedly the influence of these factors is rather large. Results about the ease to implement a changed scenario version on top of a base version

are unreliable if both versions are known to participants beforehand. However, this problem is discussed in [PKMS08]. Auditing procedures to prevent against cheating and solutions overly optimized towards the problem scenarios are provided by the code review performed at the workshops.

Accuracy 6 (Systematic Data Review) *The data collected, analyzed and presented in the course of the evaluation shall be systematically examined for possible errors.*

There have been repeatedly issues with the testbed supporting the evaluation. This was particularly critical in case of the surprise scenarios where participants had to create solutions over night and were sometimes troubled by testbed bugs without sufficient support by the organizers to fix them. The problem was aggravated by the source code of the testbed not being publicly available in its entirety. A process of reviewing the data being assembled via the server logs during the evaluation has not been implemented yet.

Accuracy 7 (Analysis of Qualitative and Quantitative Information) *Qualitative and quantitative information shall be analyzed in an appropriate, systematic way so that the evaluation questions can be effectively answered.*

The benchmark is explained in a way that everyone should be able to understand and values and limitations of the methods are discussed to some extent in [PKMS08]. However, no satisfying means for measuring the flexibility and adaptability of solutions as well as their competitive advantage over traditional programming techniques have been found yet. With this respect it is not clear whether the used change-based approach is a good measure of the fitness for purpose. A tool that may have good fitness for purpose may obtain a bad performance score if the scenario is too simple to illustrate the benefits of the tool. A tool that supposedly does not have good fitness for purpose can still obtain a good score. In fact, at the Stanford workshop a Java programmer created a solution from scratch using plain Java programming with less effort than the employed semantic technologies. It is therefore not clear, whether the scores represent the capabilities of the technologies fairly and accurately.

Accuracy 8 (Justified Conclusions) *The conclusions reached in the evaluation shall be explicitly justified so that the audiences can assess them.*

The initiative avoids making explicit conclusions in particular about the superiority or inferiority of participating technologies beyond the pure assessment of

demonstrated capabilities. However, a more thorough discussion of alternative interpretations of the controversial flexibility measures would be desirable.

Accuracy 9 (Meta-Evaluation) *The evaluation shall be documented and archived appropriately so that a meta-evaluation can be undertaken.*

The evaluation is only partially documented and not all procedures especially those performed at earlier workshop have been included. Furthermore, much of the evaluation is performed as part of demonstrations and code reviews during workshops. These have generally not been protocolled in detail. Thus, a meta-evaluation is difficult to perform. Alternative evaluations comparing the flexibility and efficiency of semantic technologies with that of traditional programming techniques have also been performed in some projects (see Section 3.1.4). No comparison with these evaluations has been prepared.

A.2. S3 Contest

Please note that the analysis of the S3 Contest represents the state as of end of 2009 but covers only the OWL-S and SAWSDL matchmaking tracks and excludes the JGDEval which was added as a complementary third track in 2009 as part of this thesis work. An assessment of JGDEval is provided in Section 8.7.

A.2.1. Utility Requirements

The Utility Requirements are intended to ensure that an evaluation is guided by both the clarified purposes of the evaluation and the information needs of its intended users.

Utility 1 (Stakeholder Identification) *Persons or groups potentially interested in or affected by the evaluation should be identified and contacted, so that their interests can be clarified and taken into consideration when designing the evaluation.*

The scope of the evaluation, potential participants and users of the evaluation results are defined. People in the community were repeatedly contacted by email, in person at conferences and through common mailing lists in the community. There were several calls for general participation and involvement in the initiative is encouraged.

Utility 2 (Clarification of the Purposes of the Evaluation) *The purposes of the evaluation shall be stated clearly, so that the stakeholders can provide relevant*

comments on these purposes, and so that the evaluation team knows exactly what it is expected to do.

The goals and purposes of the evaluations are clearly defined and the usage context described. However, even though tool improvement is one of the goals of the initiative, the evaluation results do not provide extensive discussion of the technology characteristics and development practices that lead to them. Such analysis is left to participants. The collection and publishing of corresponding papers has not been performed as part of the initiative.

Utility 3 (Evaluator Credibility and Competence) *The persons conducting an evaluation shall be trustworthy as well as methodologically and professionally competent, so that the evaluation findings achieve maximum credibility and acceptance.*

The initiative is led by an international committee of seven people from different institutions that properly reflects the diversity of the interested research community. People in the wider community have a chance to become actively involved and have been approached to do so.

Utility 4 (Information Scope and Selection) *The scope and selection of the collected information shall make it possible to answer relevant questions about the evaluand and, at the same time, consider the information needs of the client and other stakeholders.*

The design of the evaluation was discussed at public meetings at conferences and workshops several times. There are no competing evaluation approaches with a similar focus used by different groups. The problems being addressed by the evaluation are clearly defined and representatives of the problems SWS are supposed to address. However, the selection of input data is supported neither by empirical work nor a model nor theory but rather determined by the availability of current test collections. It is not clear whether this input data in its entirety is a good representative of data that the evaluated technologies are reasonably expected to handle in a natural setting. The tasks and problems of the evaluation can be scaled to be more or less complex by scaling the used test collection (within the limits of the test data available in total). The tasks can be solved and exemplary solutions exist. Poor and good solutions are possible as is evident from previous evaluation results.

Utility 5 (Transparency of Values) *The perspectives and assumptions of the stakeholders that serve as a basis for the evaluation and the interpretation of the*

evaluation findings shall be described in a way that clarifies their underlying values.

The assumptions behind the evaluation use case scenarios are clear. However, the presentation of evaluation results contains only a very brief description of the design goals and development contexts of the participating matchmakers. They are not compared in detail.

Utility 6 (Report Comprehensiveness and Clarity) *Evaluation reports shall provide all relevant information and be easily comprehensible.*

The setup of the evaluation including prerequisites, assumptions, input data, roles and tasks, information collection and data analysis is clearly described. The findings of the evaluation are comprehensible for all stakeholders and also for interested outsiders. Yet, the presentation of evaluation results is rather brief and does not comprise detailed raw data etc.

Utility 7 (Evaluation Timeliness) *The evaluation shall be initiated and completed in a timely fashion so that its findings can inform pending decision and improvement processes.*

The evaluation is performed as a continuous series of subsequent yearly evaluation events. The presentation of results is co-located with a major conference in the area, thus accommodating the constraints of as many participants as possible. Feedback about the evaluation is primarily collected during the workshop that hosts the result presentation. The evaluation has been extended and updated each time it was executed.

Utility 8 (Evaluation Utilization and Use) *The evaluation shall be planned, conducted, and reported in ways that encourage attentive follow-through by stakeholders and utilization of the evaluation findings.*

The initiative is well publicized via mailing lists, open calls for participation, word of mouth contact and has been presented at major conferences in the area several times. A web page provides comprehensive information about the initiative. Participation in the evaluation is open to all interested parties any time without any restrictions or fees. The evaluation can be executed offline but is primarily scheduled as part of a yearly evaluation event. Everything needed to execute the evaluation offline is freely available, however, the source code of the tool environment performing the actual evaluation is not available. The results of the evaluation are

well publicized and discussed at a workshop. However, the raw data underlying the consolidated evaluation results has not been made available so far.

A.2.2. Feasibility Requirements

The Feasibility Requirements are intended to ensure than an evaluation is planned and conducted in a realistic, thoughtful, diplomatic and cost-effective manner.

Feasibility 1 (Appropriate Procedures) *Evaluation procedures, including information collection procedures, shall be chosen so that the burden and cost placed on the stakeholders is appropriate in comparison to the expected benefits of the evaluation.*

The pros and cons of the chosen evaluation setup or evaluation procedures involving higher or lower effort are not discussed. The evaluation is supported by an evaluation requirement which almost entirely automates the collection and analysis of data. The effort involved in participation is thus very low.

Feasibility 2 (Diplomatic Conduct) *The evaluation shall be planned and conducted so that it achieves maximal acceptance by the different stakeholders with regard to the evaluation process and findings.*

The setup of the evaluation campaign does not explicitly encourage participants to cooperate and mutually learn from each other. The initiative is organized as a contest rather than a challenge. A winner in each category is awarded every year. To the best of our knowledge, the presentation of evaluation results is not discussed with the participants before they are made public.

Feasibility 3 (Evaluation Efficiency) *The relation between cost and benefit of the evaluation shall be appropriate.*

The participants don't need special skills, knowledge or training and the effort required in participation is minimal. There is no monetary cost involved for the participants and only minimal cost for the organizers (the traveling cost associated with the workshop where results are presented).

A.2.3. Propriety Requirements

The Propriety Requirements are intended to ensure that in the course of the evaluation all stakeholders are treated with respect and fairness, that the evaluation

achieves maximum objectivity and provides an unbiased and appropriate analysis of the technologies under examination.

Propriety 1 (Formal Agreement) *Obligations of the formal parties to an evaluation (what is to be done, how, by whom, when) shall be agreed to in writing, so that these parties are obligated to adhere to all conditions of the agreement or to renegotiate it.*

No formal agreement about the responsibilities and commitments of everyone involved in the evaluation is prepared, but the responsibilities and commitments of everyone are public and transparent from the description of the initiative.

Propriety 2 (Protection of Individual Rights) *The evaluation shall be designed and conducted in a way that protects the welfare, dignity and rights of all stakeholders.*

There is no process for users to vet their results before they are released but users may veto the release of their results or withdraw their entry to the evaluation.

Propriety 3 (Complete and Fair Investigation) *The evaluation shall undertake a complete and fair examination and description of strengths and weaknesses of the evaluand so that strengths can be built upon and problem areas addressed.*

Evaluation results are presented as consolidated scores and charts and thus provide only limited information about the causes of the observed performance of the tools or methods under investigation. However, the charts allow a meaningful comparison of evaluated technologies, even though with limited detailedness. The coverage of the characteristics of technologies is limited by the availability of test data. The selection of test data is not justified but rather a matter of fact based upon the data's availability.

Propriety 4 (Unbiased Conduct and Reporting) *The evaluation shall take into account the different views of the stakeholders concerning the evaluand and the evaluation findings. Like the entire evaluation process, the evaluation report shall evidence the impartial position of the evaluation team. Value judgments shall be made as unemotionally as possible.*

It is not entirely clear whether the evaluation is completely independent from particular solutions since most of the test data has been developed in the context of particular solutions. The evaluation is not tied to a particular platform but

based upon the availability of test data tied to particular semantic formalisms. It is thus only applicable to a subset of the available approaches. SAWSDL and OWL-S matchmakers are covered but the evaluation is not applicable to other approaches, in particular not to WSML matchmakers. The evaluation can be applied to research prototypes as well as mature products and tasks or inputs can be easily down-scaled in size. The evaluation report does not discuss potential biases or aspects that may be conceived as such by a stakeholder. There is also no reviewing process for the evaluation results and report in place.

Propriety 5 (Disclosure of Findings) *As far as possible, all stakeholders shall have access to the evaluation findings.*

The scores and evaluation results are clearly documented but the raw data underlying the results is not made available. Thus, evaluation results are not fully reproducible without access to the evaluated matchmaker entries. The terms of publication of evaluation findings, how, when, by whom, according to which criteria and with respect to which limitations and restrictions are not defined in much detail as part of the call for participation but there is an established procedure how results are published from previous years.

A.2.4. Accuracy Requirements

The accuracy requirements are intended to ensure that an evaluation produces and discloses valid, accurate, precise, reliable and useful information and findings pertaining to the evaluation purposes and questions.

Accuracy 1 (Description of the Evaluand) *The evaluand shall be described and documented clearly and accurately so that it can be unequivocally identified.*

The benchmark clearly identifies the tasks to perform and the characteristics of a technology under investigation. The quality criteria being assessed in the evaluation are clearly defined and the tools or techniques that are intended to be evaluated specified. The evaluation requirements and assumptions are provided.

Accuracy 2 (Context Analysis) *The context of the technologies being evaluated shall be examined and analyzed in sufficient detail.*

The contexts of the technologies being evaluated are described only very briefly in the evaluation report. A more comprehensive coverage would be desirable.

Accuracy 3 (Described Purposes and Procedures) *Object, purposes, methodology and procedures of an evaluation, including the applied methods, shall be accurately documented and described so that they can be identified and assessed.*

The measured characteristics and evaluation criteria as well as the tasks to perform and the input data are clearly defined in the benchmark specification. The procedure how the evaluation is executed and how the resulting scores are compiled are documented properly and in sufficient detail. The infrastructure and software supporting the data collection and analysis is publicly available and sufficiently documented.

Accuracy 4 (Disclosure of Information Sources) *The information sources used in the course of the evaluation shall be documented in appropriate detail so that the reliability and adequacy of the information can be assessed.*

The evaluation setup clearly specifies how all data is obtained but unfortunately, the raw data collected during the evaluation is not accessible.

Accuracy 5 (Valid and Reliable Information) *The data collection procedures shall be chosen and developed and then applied in a way that ensures the reliability and validity of the data with regard to answering the evaluation questions. The technical criteria shall be based on the standards of quantitative and qualitative social research.*

Assumptions made by the evaluation regarding the expected user and usage context are realistic, but the selection of performance measures is not discussed or supported by empirical work. Different persons applying the evaluation on the same technology twice would get the exact same results. The results are not affected by unpredictable environment behavior. However, other threats to the evaluation's validity, in particular regarding the influence of the characteristics of the used test collections and the performance measures used are not discussed. The runtime performance measures are expected to be significantly affected by the maturity of implementation. Since the test collections used by the evaluation are publicly available, runtime performance as well as retrieval correctness results may be affected by optimizing a technology for the used data. No procedures for auditing against cheating or overly optimized solutions are in place.

Accuracy 6 (Systematic Data Review) *The data collected, analyzed and presented in the course of the evaluation shall be systematically examined for possible errors.*

There have been reports of bugs in the supporting infrastructure although these bugs were not found to affect the evaluation results. Unfortunately, the source code of the test environment is not publicly available. Therefore, its correctness can not be verified by interested stakeholders. There is no reviewing process in place which critically reviews all data being assembled during the evaluation.

Accuracy 7 (Analysis of Qualitative and Quantitative Information) *Qualitative and quantitative information shall be analyzed in an appropriate, systematic way so that the evaluation questions can be effectively answered.*

The measures and formulae used to analyze evaluation results are clearly specified and explained in a way that everyone can understand. However, the values and limitations of the methods and measures used are not discussed explicitly. The compiled measures used in the evaluation are believed to be good indicators of the fitness for purpose of the evaluated technologies. However, some known restrictions and weaknesses of the measure have not been investigated. It is not entirely clear whether a tool that does have fitness for purpose may obtain a bad performance score and vice versa. Besides these restrictions, the computed scores represent the capabilities of a single technology fairly and accurately and can be used to directly compare two technologies.

Accuracy 8 (Justified Conclusions) *The conclusions reached in the evaluation shall be explicitly justified so that the audiences can assess them.*

Conclusions are made very cautiously and primarily left to participants. Their scope as well as their limitations is defined and discussed.

Accuracy 9 (Meta-Evaluation) *The evaluation shall be documented and archived appropriately so that a meta-evaluation can be undertaken.*

The evaluation is documented in sufficient detail to perform a meta-evaluation of the evaluation approach. However, the actual evaluation can not be validated without access to the raw data collected during the evaluation and all evaluated tools. No meta-evaluations and comparisons with previous or alternative evaluations (e.g., using alternative measures available from information retrieval) have been performed.

A.3. WS Challenge

Please note that the analysis of the WS Challenge represents the state as of end of 2009 and covers editions prior to 2008 only to a very limited extent since almost no information about these editions is still available online (see Section 3.1.3).

A.3.1. Utility Requirements

The Utility Requirements are intended to ensure that an evaluation is guided by both the clarified purposes of the evaluation and the information needs of its intended users.

Utility 1 (Stakeholder Identification) *Persons or groups potentially interested in or affected by the evaluation should be identified and contacted, so that their interests can be clarified and taken into consideration when designing the evaluation.*

The scope of the evaluation, potential participants and users of the evaluation results are defined. People in the community were repeatedly contacted through common mailing lists. There were several calls for participation and involvement in the benchmark design was supposedly possible through discussion at the evaluation events.

Utility 2 (Clarification of the Purposes of the Evaluation) *The purposes of the evaluation shall be stated clearly, so that the stakeholders can provide relevant comments on these purposes, and so that the evaluation team knows exactly what it is expected to do.*

The goals and purposes of the evaluations are clearly defined and the usage context described. Unfortunately, evaluation results do not provide insights about the technology characteristics and development practices that lead to them and also do not provide explicit technology improvement recommendations.

Utility 3 (Evaluator Credibility and Competence) *The persons conducting an evaluation shall be trustworthy as well as methodologically and professionally competent, so that the evaluation findings achieve maximum credibility and acceptance.*

The evaluation team consists of people from two universities. Even though their trustworthiness and competence is not in question, a larger organizing committee better reflecting the diversity of the research community is preferable. Supposedly, people in the wider community have the option of becoming actively involved and the organizing committee has in fact changed over time.

Utility 4 (Information Scope and Selection) *The scope and selection of the collected information shall make it possible to answer relevant questions about the evaluand and, at the same time, consider the information needs of the client and other stakeholders.*

It is not clear whether the design of the evaluation was discussed but it can be expected that this was the case at the evaluation events when results were presented. The scope of the evaluation is clearly defined and the problems addressed by the evaluation believed to be valid representatives of problems found in actual practice. However, the input data is artificially generated and its characteristics not supported by a model nor theory. The test data generator supports the generation of arbitrarily complex or large test corpora, thus allowing to easily scale the evaluation. The evaluation tasks can be solved as is evident from previous evaluation results. These also illustrate that good as well as poor solutions are possible.

Utility 5 (Transparency of Values) *The perspectives and assumptions of the stakeholders that serve as a basis for the evaluation and the interpretation of the evaluation findings shall be described in a way that clarifies their underlying values.*

The assumptions behind the evaluation use case are described but the design goals of the evaluated technologies are not described nor compared in detail. However, there are short papers accompanying the evaluation entries where participants can provide this information. Unfortunately, these papers are not linked from the evaluation web page where results are presented.

Utility 6 (Report Comprehensiveness and Clarity) *Evaluation reports shall provide all relevant information and be easily comprehensible.*

The setup of the evaluation including prerequisites, assumptions, input data, roles and tasks, information collection and data analysis is clearly described, except for the architectural award whose criteria are not made available. The findings of the evaluation are comprehensible for all stakeholders and also for interested outsiders, but difficult to interpret with respect to their underlying causes. They are reported very briefly and condensed. In fact, performance scores are only available for the top performing technologies. More comprehensive evaluation reports are desirable here.

Utility 7 (Evaluation Timeliness) *The evaluation shall be performed in a timely fashion so that its findings can inform pending decision and improvement processes.*

The evaluation is performed as a yearly event since 2005. The evaluation events are co-located with a major conference in the area such that participants can plan long ahead. The evaluation has been evolved every year.

Utility 8 (Evaluation Utilization and Use) *The evaluation shall be planned, conducted, and reported in ways that encourage attentive follow-through by stakeholders and utilization of the evaluation findings.*

The initiative is well established, has been publicized and presented several times. However, web pages about the evaluation are scattered and the pages for previous editions not online anymore. Participation in the evaluation is open to all interested parties and free of any fee or other restriction. The evaluation is performed on a yearly basis, but additionally, is can also be executed offline. All materials needed for offline execution are publicly available. The results of the evaluation were presented at a major conference but details and the underlying raw data are not available. Generally, evaluation results are presented with too little detail, making the utilization and further analysis of evaluation results by stakeholders rather difficult.

A.3.2. Feasibility Requirements

The Feasibility Requirements are intended to ensure than an evaluation is planned and conducted in a realistic, thoughtful, diplomatic and cost-effective manner.

Feasibility 1 (Appropriate Procedures) *Evaluation procedures, including information collection procedures, shall be chosen so that the burden and cost placed on the stakeholders is appropriate in comparison to the expected benefits of the evaluation.*

The evaluation discusses pros and cons of the used test data generator and also the evolution of the evaluation setup over years. Comprehensive tool support for the evaluation is readily accessible and the execution of the evaluation, including the analysis of the collected data is largely automated.

Feasibility 2 (Diplomatic Conduct) *The evaluation shall be planned and conducted so that it achieves maximal acceptance by the different stakeholders with regard to the evaluation process and findings.*

The evaluation is organized as a contest and awards winners and runner-ups. In fact, performance results are only reported for the top performing technologies. The evaluation does not make the impression of encouraging participants and other

stakeholders to cooperate and mutually learn from each other. It is not clear whether the presentation of evaluation results is discussed with the participants before they are made public.

Feasibility 3 (Evaluation Efficiency) *The relation between cost and benefit of the evaluation shall be appropriate.*

The participants in the evaluation do not need special skills, knowledge or training. The minimal and optimal time required for participation is not discussed in the evaluation setup, but due to the large degree of automation the effort can be expected to be rather low. However, participants are required to physically attend the evaluation session in order to present their systems. This involves cost for traveling and conference registration. Other than that there is no monetary cost involved in participation in the evaluation. It might be desirable to alternatively offer an offline evaluation as the in some ways similar S3 Contest does.

A.3.3. Propriety Requirements

The Propriety Requirements are intended to ensure that in the course of the evaluation all stakeholders are treated with respect and fairness, that the evaluation achieves maximum objectivity and provides an unbiased and appropriate analysis of the technologies under examination.

Propriety 1 (Formal Agreement) *Obligations of the formal parties to an evaluation (what is to be done, how, by whom, when) shall be agreed to in writing, so that these parties are obligated to adhere to all conditions of the agreement or to renegotiate it.*

To the best of our knowledge there does not seem to be a formal agreement about the responsibilities and commitments of everyone involved in the evaluation. The responsibilities of the single organizers are not evident and transparent from the evaluation description.

Propriety 2 (Protection of Individual Rights) *The evaluation shall be designed and conducted in a way that protects the welfare, dignity and rights of all stakeholders.*

It does not appear as if there was a process for users to vet their results before they are released. It is not clear whether users can veto the release of their results.

Propriety 3 (Complete and Fair Investigation) *The evaluation shall undertake a complete and fair examination and description of strengths and weaknesses of the evaluand so that strengths can be built upon and problem areas addressed.*

The evaluation results in a few single scores and does not provide detailed information about the performance of the tool under investigation or the causes of the measured performance behavior. Results are only reported for the top four performers and for those only very briefly. They do not really make the strengths and weaknesses of the evaluated technologies explicit. A detailed comparison is difficult based upon the provided evaluation results. With this respect, the WS Challenge is really more of a contest than an investigative performance evaluation.

Propriety 4 (Unbiased Conduct and Reporting) *The evaluation shall take into account the different views of the stakeholders concerning the evaluand and the evaluation findings. Like the entire evaluation process, the evaluation report shall evidence the impartial position of the evaluation team. Value judgments shall be made as unemotionally as possible.*

The evaluation is independent from particular solutions. The current organizers used to be participants in the evaluation but quit participation once they took over the organization of the initiative. The evaluation is not tied to a particular platform or technology and its tasks and data are specified in a way that ensures its applicability to different technologies without being biased towards specific ones. The evaluation can be used for all applicable technologies in the field and applied to research prototypes as well as mature products, even though the measures are probably highly dependent from optimizations of the running code. The test data generator supports the provisioning of inputs at different sizes and complexities. Unfortunately, the evaluation report does not discuss potential biases. Furthermore, there does not seem to be a reviewing process for evaluation results and reports in place.

Propriety 5 (Disclosure of Findings) *As far as possible, all stakeholders shall have access to the evaluation findings.*

To the extent that they are provided, the scores and evaluation results are clearly documented except for the architectural award that is not motivated nor justified. However, as mentioned above, evaluation results are only available for the top performing entries. Additionally, the underlying raw data is generally not published such that the reproducibility of evaluation findings is limited. It is not clear

whether the evaluation findings have been disclosed according to some previously agreed upon terms.

A.3.4. Accuracy Requirements

The accuracy requirements are intended to ensure that an evaluation produces and discloses valid, accurate, precise, reliable and useful information and findings pertaining to the evaluation purposes and questions.

Accuracy 1 (Description of the Evaluand) *The evaluand shall be described and documented clearly and accurately so that it can be unequivocally identified.*

The benchmark clearly identifies the tasks to perform and the characteristics of a technology under investigation. The quality criteria being assessed in the evaluation are clearly defined and the tools or techniques that are intended to be evaluated specified. The evaluation requirements and assumptions are provided.

Accuracy 2 (Context Analysis) *The context of the technologies being evaluated shall be examined and analyzed in sufficient detail.*

The contexts of the evaluated technologies are not described in the evaluation report and their potential influence on the evaluation findings is not discussed. However, participants provide a four page technical description of their entry where corresponding information can be provided. Unfortunately, these papers are not linked from the evaluation web page.

Accuracy 3 (Described Purposes and Procedures) *Object, purposes, methodology and procedures of an evaluation, including the applied methods, shall be accurately documented and described so that they can be identified and assessed.*

The measured characteristics and evaluation criteria as well as the tasks to perform and the input data are clearly defined in the benchmark specification. The procedure how the evaluation is executed and how the resulting scores are compiled is documented properly and in sufficient detail. The infrastructure and software supporting the data collection and analysis is publicly available and also well documented.

Accuracy 4 (Disclosure of Information Sources) *The information sources used in the course of the evaluation shall be documented in appropriate detail so that the reliability and adequacy of the information can be assessed.*

The evaluation setup clearly specifies how all data is obtained but unfortunately, the raw data collected during the evaluation is not accessible.

Accuracy 5 (Valid and Reliable Information) *The data collection procedures shall be chosen and developed and then applied in a way that ensures the reliability and validity of the data with regard to answering the evaluation questions. The technical criteria shall be based on the standards of quantitative and qualitative social research.*

Assumptions made by the evaluation are described. The selection of performance measures is explained and justified, even though supported neither by empirical work nor a model nor theory. Different people applying the evaluation on the same technology twice would very likely get the same results. Results are not expected to be affected by unpredictable environment behaviors. Threats to the evaluation's validity like those resulting from special characteristics of the generated test data are not discussed in depth. Since runtime performance measures are generally highly affected by code optimizations and caching techniques, it has to be suspected that the evaluation is vulnerable to extensive use of optimization and that results are affected by the maturity of implementation of a particular technique under investigation. There do not seem to be procedures for auditing evaluated technologies to identify solutions that are overly optimized towards the procedure and measures of the evaluation.

Accuracy 6 (Systematic Data Review) *The data collected, analyzed and presented in the course of the evaluation shall be systematically examined for possible errors.*

The infrastructure supporting the evaluation seems to run in the expected way, but test procedures and results have not been documented. The infrastructure is fully available, but the source code is not provided together with the binaries. Therefore, its correctness can not be verified by interested stakeholders. There is no reviewing process in place which critically reviews all data being assembled during the evaluation.

Accuracy 7 (Analysis of Qualitative and Quantitative Information) *Qualitative and quantitative information shall be analyzed in an appropriate, systematic way so that the evaluation questions can be effectively answered.*

The benchmarks and scores, except for the architectural award, are explained in a way that everyone can understand. The compiled measures are deemed to be good indicators of the performance of at least the implementation under investi-

gation. However, an unoptimized implementation of a technique that in principle has fitness for purpose may receive a poor performance score. It is also not clear whether the computed scores are vulnerable to a great variability depending on the actual input challenge (test data) being used. A corresponding discussion would be desirable. Otherwise the computed scores seem to represent the capabilities of a single implementation fairly and accurately and can be used to directly compare two implementations.

Accuracy 8 (Justified Conclusions) *The conclusions reached in the evaluation shall be explicitly justified so that the audiences can assess them.*

The evaluation awards winners and runner-ups but otherwise does not draw any conclusions. The scope and limitations of the provided results (for instance, its dependency on the actual test data used) are not discussed.

Accuracy 9 (Meta-Evaluation) *The evaluation shall be documented and archived appropriately so that a meta-evaluation can be undertaken.*

The key purposes, steps, methods, data and findings of the evaluation are available, even though only in its consolidated form and not as raw data. Furthermore, the data from editions prior to 2008 does not seem to be available anymore. To the best of our knowledge, no meta-evaluations and comparisons with previous or alternative evaluations have been performed.

APPENDIX B

Additional Information on the Functional Scope Benchmark

This appendix makes some additional information about the benchmark for assessing the functional scope of SWS frameworks presented in Chapter 6 available for reference. The following Section B.1 provides a summary of the detailed comparison of approaches to the benchmark problems. Section B.2 presents an assessment of the benchmark with respect to the evaluation requirements specified in Section 4.2.

B.1. Detailed Solution Comparisons

In addition to the pure evaluation results described in Section 6.6, in-depth comparisons of different technologies have been performed based upon the solutions to the problem scenarios of the benchmark. These comparisons were jointly written by the authors of the compared solutions [KTZ+07, KTKR+08, KKRMS08, KVW+08]. This project greatly increased the mutual understanding for each other technologies and the tradeoffs involved in them.

In other words: "[...] each of these comparison chapters involved a great deal of joint analysis by teams working with different technologies, and because the teams had to agree on the analysis. However, exactly for those reasons, these comparison chapters will be particularly valuable to readers attempting to understand the technical issues and solutions" [Pet08].

Unlike the evaluation results that focus on which problems were solved, the comparisons focus on *how* the problems were solved. Therefore, the comparison criteria are differently organized than the evaluation results. The comparison is provided according to the following criteria: underlying technology, service descriptions, goal descriptions, data model (ontologies), matchmaking, preferences and ranking, dy-

namic properties and service execution. The findings of each category will be briefly summarized below, for the complete coverage the interested reader is referred to [KTZ+07, KTKR+08, KKRMS08, KVW+08].

Please note that the detailed comparisons have been performed in 2007. Therefore, the comparisons do not cover some of the teams that joined the SWS Challenge more recently and are restricted to the joint team from Politecnico Milano and Cefriel, the DERI team, the joint team from University of Dortmund and University of Potsdam and the University of Jena team (for brevity, in the following the corresponding solutions will be referred to as Milano, DERI, Potsdam and Jena solution respectively). Furthermore, the Logistics Scenario had not yet been added to the SWS Challenge testbed at that time and is thus also not covered by the comparisons.

B.1.1. Underlying technologies

The Milano solution is based on SWE-ET (Semantic Web Engineering - Environment and Tools) which combines two technologies. Service discovery is performed by the GLUE web service discovery engine which is based on F-logic. Service invocation and data mediation tasks are performed by the WebRatio framework [TVCF08, BCV+08].

The DERI solution is based upon the WSMX (Web Service Execution Environment) services framework. Internally, the IRIS Datalog reasoner is used to reason about the semantic service descriptions. WSMX readily includes components that handle the invocation of services within the framework [ZMV08].

The Potsdam solution again combines two technologies. The matchmaking aspects are covered by the miAamics personalization framework which is embedded into the jABC platform that manages the overall discovery process and also handles all service invocation and data mediation tasks [KMS+08].

Finally, the Jena solution is based upon the DIANE framework, a technology specifically developed to automate the whole service consumption lifecycle. The DIANE framework employs a custom reasoning operation for service discovery which is directly implemented within the framework, i.e. without leveraging a standard logic reasoner. Service invocation and data mediation tasks are also handled by native components of the framework [KKR08c].

B.1.2. Service descriptions

The Milano solution service descriptions follow the WSMO modeling approach but extend it to clearly separate between web service classes and instances. The previous define the schema to describe a web service in a certain domain. The web service class for the shipping services, for instance, defines properties like *operation range*, *price* or *weight limit*. Web service instances instantiate this schema with concrete

property values, like the United States as operation range and a weight limit of 50 lbs. All descriptions are specified in F-logic.

The DERI solution also relies on the WSMO modeling approach. Services are formalized using the WSML Flight language fragment. In contrast to the Milano descriptions, which primarily serve as data containers, the DERI web service descriptions not only define properties, but also axioms that actually perform part of the matchmaking, like an *isShipped* axiom that computes whether a service is applicable for shipping a given package or axioms encoding the rules to compute the price of a given shipment. Unlike the Milano solution, DERI does not distinguish between web service classes and instances. The definition of the necessary domain schema is entirely done in corresponding domain ontologies.

The Potsdam solution treats services as a record in an underlying database, i.e., as a collection of property values. This is not so much different from the Milano solution. However, unlike the Milano solution that is based on F-logic and can thus process numerical values directly, miAamics is based on Boolean properties (called attributes or categories). Therefore, numerical property values are abstracted to Boolean properties via criteria that specify whether or not an attribute corresponds to a concrete offer. For instance, instead of specifying a concrete package weight limit of 50 lbs, a corresponding service is characterized within miAamics as a service capable of shipping *medium heavy* packages using a corresponding rule.

The Jena approach employs a description formalism called DSD [KKRM05]. Within this formalism, services are modeled as the set of concrete effects that they can offer by specifying conditions on domain properties defined in domain ontologies. A shipping service, for instance, can be characterized as being able to provide the set of shipments, where the weight property is smaller than 50 lbs, the destination address is located in a specified set of countries or continents etc. Therefore, the Jena approach explicitly provides some matchmaking rules (e.g., weight smaller than 50 lbs) in the web service descriptions. This is different from the Milano (and Potsdam) solution, where service descriptions specify values (like 50 for the weight limit property), but the interpretation of these values (an upper limit on the weight of the package in the example) is specified elsewhere. It is similar to the DERI approach but much less expressive. The Jena approach only allows the specification of simple conditions on attributes whereas the DERI approach supports arbitrarily complex logic rule specifications within the service descriptions.

B.1.3. Goal descriptions

Following the WSMO modeling approach, the Milano approach distinguishes between services and goals, but otherwise treats goals very similar to web services. Goal classes define a schema for the property of goals in a certain domain and goal

instances instantiate this schema with concrete property values. As for the service descriptions, F-logic is used to formalize the goals.

Similar to the Milano solution, the DERI solution also distinguishes between goal and service descriptions. The goal descriptions are similar in nature to the service descriptions and also specified in WSML Flight. In the DERI solution, the service-side matchmaking rules specified in the web service descriptions are complemented by request-side matchmaking rules specified in the goal's postconditions. A shipping goal, for instance, defines the concrete values of a desired shipment and a postcondition that states that the before mentioned isShipped axiom must hold (the shipper is suitable) and the price of the shipment must be smaller than a certain limit.

Within the Potsdam approach, goals, similarly to services, are stored as entries in a database and characterized by a set of Boolean predicates that abstract from numerical properties and categorize the requests into classes. An example of such a predicate is that a package weighing more than 50 lbs is categorized as a medium heavy shipment. Other than in the Milano solution, matchmaking specifications are independent from the pure request specifications and will be covered further below.

The Jena approach also distinguishes between service request and offer descriptions. As for the offer descriptions, DSD is used to formalize the goals. Within this approach requests are modeled as the set of concrete effects that are acceptable to the requester. Just like for the modeling of service offers, the acceptable effects are characterized using restrictions on properties of the domain ontology, e.g. by specifying that the weight of the parcel to be shipped must be exactly 50 lbs and the price must be less than 20 $. Unlike offer descriptions, request descriptions are based on fuzzy instead of crisp effect sets. This allows expressing user preferences and will be covered further below.

B.1.4. Data model

Despite of largely different underlying formalisms (F-logic, WSML Flight, DSD), the domain ontologies of the Milano, DERI and Jena solutions looked rather similar. This was probably due to the fact that the modeled domains were limited in size and complexity and therefore did not require the specification of complex axioms, relations or rules within the data model.

The Potsdam approach differed more significantly. Unlike all other approaches, the Potsdam data model comprised a categorization of the property space of the domain of interest into a discrete taxonomy by means of rules. This taxonomy served as the basis for the before mentioned Boolean predicates necessary for the modeling of web services and goals.

B.1.5. Matchmaking

Not surprisingly, approaches differed most significantly within the concrete process of the matchmaking between offers and requests. Please note that mediating between goals and web services potentially requires two tasks. On the one hand, the supply (as expressed by the available offers) has to be compared with the demand (as specified in the goal) in terms of provided and requested functionality. This is covered in the following. On the other hand, goals and web services may be expressed using different ontologies in which case a semantic alignment has to be performed additionally to the functional matching. This was not required by the Shipment and Hardware Purchasing Scenario covered by the solution comparisons but only later added within the Logistics Scenario and is therefore not covered here.

The matchmaking within the Milano and DERI solution was fairly similar. Both approaches are based on rule languages (WSML Flight and F-logic) and therefore directly modeled the necessary matchmaking rules as filters on the available services. All reasoning operations were performed with standard logic reasoners (the IRIS Datalog reasoner and the Flora-2 reasoner) which both directly support arithmetic computations and custom functions. As indicated above, the DERI approach covered service-side rules in the service descriptions (e.g., restrictions on the geographic coverage) and requester-side rules in the request descriptions (e.g., restrictions on the price). In contrast, the Milano approach emphasized the notion of a web-service-goal-mediator (wgmediator), an independent entity that links web services with goals and contains all necessary filter rules. Some further differences regarding preferences and ranking are covered in the corresponding section below.

The Potsdam solution leveraged a personalization engine for matchmaking and therefore differed significantly from the other approaches. As mentioned, services and goals within this approach are characterized via a set of Boolean predicates. The evaluation of complex rules, like the computation of shipping prices from the weight and destination of a package is performed in a pre-processing by jABC independent from the pure matchmaking. The pure matchmaking starts when all properties have been computed and are locally available for comparison. It is based upon, potentially different, strategies, which are basically a set of rules to employ. Rules consist of a premise (an attribute of a goal, like *destination is UK*), a conclusion (an attribute of an offer, like *ships to UK*) and a weight. If the premise holds for a request and the conclusion holds for an offer, the weight of the rule is added to the matching score of the corresponding offer request pair. This matching score is then used to filter and rank the available services. Notably, the discretization of numerical values into Boolean predicates resulted in a certain loss of precision and discrimination power during the matchmaking, but also superior runtime performance.

The Jena solution matchmaking is based on a special reasoning operation called *subset*. Subset computes the membership degree of the highest scoring element of

an offer description within a given goal description, corresponding to answering the question how well an offer fits to what the requester has specified as acceptable. Recall that offers are described as the set of effects that a service can provide whereas requests are described as the fuzzy set of effects acceptable to the requester to different degrees. Services and goals need to be specified with respect to a common domain ontology. Subset is implemented as a Java matchmaker that performs a recursive comparison of the attributes of a request with the corresponding attributes from the offer. During this comparison, where possible the offer is automatically configured optimally by choosing appropriate input values for the web service. To keep the matchmaking efficient, the Jena approach supports only a certain set of attribute conditions and does not support arbitrarily complex rules. Therefore, the computation of the shipment price from the destination address and the weight of a package, for instance, had to be delegated to external web services. From a conceptual point of view, the Jena solutions combines provider-side restrictions from the offer descriptions with client-side restrictions from the request descriptions with domain independent general matchmaking rules as implemented by *subset*.

Overall, the Milano and DERI approach offer the greatest flexibility since they allow specifying arbitrary matchmaking rules in the wgmediator respectively the offer and goal descriptions. On the other hand, they require the modeling of a concrete rule for each attribute that needs to be compared which leads to increasingly complex descriptions in scenarios with many attributes to be compared (like the Hardware Purchasing Scenario). In contrast, the Jena and Potsdam approach are more restricted with respect to the comparisons they may perform and, in case of the Jena approach, combine domain specific matchmaking rules with reusable domain independent matchmaking rules. This results in simpler service descriptions and better runtime performance, but is paid by the disadvantage that more complex rules (like the shipping price computations) need to be performed as an additional preprocessing or delegated to external computation services.

B.1.6. Preferences and ranking

The Milano approach offers only limited support for preference based rankings (this has been improved meanwhile). Within the wgmediator, several discrete matching levels may be defined that each corresponds to a number of rules or filters that are applied on the candidate services. This is very similar to the Potsdam approach where each rule is associated with a weight and a service retrieves the cumulated weight of all matching rules. This is a simple and efficient approach, but allows only for a finite number of discrete matching levels.

The DERI approach uses the notion of non-functional properties to specify ranking criteria. More precisely, non-functional properties may specify variables from the service's post-conditions together with an ordering (ascending or descending).

Variables are bound to concrete values during the reasoning process and the services are then ranked according to the specified ordering. This allows to rank services based upon numerical properties (like the price) without requiring a discretization of the underlying value, but does not allow for arbitrary combinations of different ranking criteria.

With respect to preferences, the Jena approach offers the most powerful support. As mentioned, requests are described as the fuzzy set of acceptable service effects. The greater the membership of a concrete service in the fuzzy request set, the higher the preference of the requester for this service. The fuzzy sets are primarily built by means of so called *direct conditions* and *connecting strategies*. Direct conditions allow to specify fuzzy conditions on single attributes, like price less than 100$, the smaller, the better. Connecting strategies allow combining the match scores from different attributes (like the price and the HDD size of a notebook) by means of specified arithmetic formulas (like weighted sum, product, etc.). This supports the expressive modeling of fine-grained preferences, even though the specification of suitable values and formulas is not always trivial.

B.1.7. Dynamic properties

Sometimes, properties of services can not be statically specified but must be dynamically retrieved. This regarded the price of one of the services in the Shipping Scenario and the currently available products in the Hardware Purchasing Scenario. The solution to this problem was relatively similar in all four approaches.

Offer properties that need to be obtained dynamically are tagged or otherwise marked. During the matchmaking process, the services are then invoked to obtain the corresponding values and the service descriptions or knowledge bases are temporarily updated with the retrieved values. In case of the DERI and the JENA approach, this process is directly supported by the corresponding framework. The Potsdam and Milano approaches, which are based on combining different technologies for matchmaking and service invocation, delegate this step the their service invocation engines (jABC respectively WebRatio). The Milano, DERI and Jena approach have been optimized to obtain dynamic data only for those services that match based on the static offer attributes. However, this optimization could be easily added to the Potsdam approach, too.

B.1.8. Service execution

In case of the Milano and Potsdam approach, data mapping and service execution is performed independent from the matchmaking via WebRatio and jABC, powerful model-based web service environments. In case of the DERI approach, the Communication Manager and Invoker components integrated into the WSMX environment

handle these tasks. Similar although less sophisticated components are provided as part of the DIANE framework leveraged by the Jena solution. All approaches use more or less declaratively specified mapping rules to automatically transform data between the semantic and syntactic (XML) level, are able to automatically compose the proper messages, invoke the web services, interpret the results and make the contained information available to the semantic level. In case of WebRatio, jABC and WSMT (the modeling environment for WSMX), quite sophisticated tool support to define the necessary mappings and to monitor the execution is available, while the Jena approach focuses on the discovery process itself and does not offer comparable tool support.

B.2. Assessment with Respect to Evaluation Requirements

In this section, an analysis of the Functional Scope Benchmark with respect to the evaluation requirements identified in Section 4.2 is provided. Each requirement is briefly stated and a short statement with respect to the questions operationalizing the requirement is given. For improved legibility, the questions are not repeated but are available in Section 4.2. The analysis presented here served as the basis for the corresponding summary in Section 8.6.2.

B.2.1. Utility Requirements

The Utility Requirements are intended to ensure that an evaluation is guided by both the clarified purposes of the evaluation and the information needs of its intended users.

Utility 1 (Stakeholder Identification) *Persons or groups potentially interested in or affected by the evaluation should be identified and contacted, so that their interests can be clarified and taken into consideration when designing the evaluation.*

The scope of the evaluation, potential participants and users of the evaluation results are defined. People in the community were repeatedly contacted by email, in person at conferences and through common mailing lists in the community. Some probe contacts with industrial players in the field (e.g., SAP) have not indicated high interest and have thus not been further pursued. There were several calls for general participation and involvement in the benchmark design was encouraged and possible through associated mailing lists and discussion at workshops.

Utility 2 (Clarification of the Purposes of the Evaluation) *The purposes of the evaluation shall be stated clearly, so that the stakeholders can provide relevant comments on these purposes, and so that the evaluation team knows exactly what it is expected to do.*

The goals and purposes of the evaluations are clearly defined and the usage context described. Evaluation results provide insights about the technology characteristics and development practices that lead to them via the papers describing the solutions to the benchmark problems and especially via the additional in-depth comparisons of solutions that have been prepared. Results do not provide explicit technology improvement recommendations because such interpretation is left to the participants. However, improvement of the evaluated tools or techniques is an explicit goal of the evaluation.

Utility 3 (Evaluator Credibility and Competence) *The persons conducting an evaluation shall be trustworthy as well as methodologically and professionally competent, so that the evaluation findings achieve maximum credibility and acceptance.*

According to the benchmark methodology, the evaluation team comprises the whole set of participants at each evaluation workshop. The general organizing team of the initiative comprises roughly fifteen people from seven institutions and thus reflects the diversity of the interested research community. The organization team is open for new people anytime and in fact has changed over time.

Utility 4 (Information Scope and Selection) *The scope and selection of the collected information shall make it possible to answer relevant questions about the evaluand and consider the information needs of the client and other stakeholders.*

The design of the evaluation was discussed at several open workshops and also within a W3C Incubator activity. To the best of our knowledge, there are no competing evaluation approaches for assessing the functional scope of SWS discovery frameworks. The problem scenarios defined by the benchmark are believed to be representative problems for SWS based discovery and matchmaking. However, they are somewhat visionary and thus not yet found in actual practice. While one of the scenarios originates from an industrial case study and the selection of problem scenarios is supported by a community approval process, it is not supported by empirical work (besides the mentioned case study) nor a model nor theory. The benchmark problems provide tasks at different complexity, all of which can be solved by existing conventional technology. Yet the task sample is hard which is demonstrated by none of the SWS approaches having been able to solve all problem levels.

Utility 5 (Transparency of Values) *The perspectives and assumptions of the stakeholders that serve as a basis for the evaluation and the interpretation of the evaluation findings shall be described in a way that clarifies their underlying values.*

The assumptions behind the evaluation use case scenarios are described in the problem scenario definitions. The design goals of the evaluated technologies are described in detail in the papers accompanying every benchmark entry. Furthermore, the detailed comparison of solutions includes a comparison of these design goals.

Utility 6 (Report Comprehensiveness and Clarity) *Evaluation reports shall provide all relevant information and be easily comprehensible.*

The setup of the evaluation regarding prerequisites, assumptions, input data, roles and tasks is clearly described. The process of the information collection, i.e., the actual certification procedure, is described in detail in a W3C Incubator Group report [PKMS08]. However, information regarding this procedure is somewhat scattered over various web sites and the mentioned document. While the findings of the evaluation are completely comprehensible for participants in the benchmarking, they may not always be easily comprehensible for the interested outsiders. Documentation of the procedure and results of the information could be improved with this respect. However, the recently added problem scenario (Logistics Management) significantly improves over the older scenarios with this respect.

Utility 7 (Evaluation Timeliness) *The evaluation shall be initiated and completed in a timely fashion so that its findings can inform pending decision and improvement processes.*

The benchmark is intended to be used frequently and embedded in a continuous benchmarking initiative which has held at least yearly workshops since 2006. Scheduling of subsequent workshops is planned during workshops in a way that tries to accommodate the preferences of everybody involved and potential new participants. There are discussion slots at every workshop where feedback about the evaluation is collected. The benchmark has been updated several times with the addition of new problem scenarios or the improvement of existing ones.

Utility 8 (Evaluation Utilization and Use) *The evaluation shall be planned, conducted, and reported in ways that encourage attentive follow-through by stakeholders and utilization of the evaluation findings.*

The SWS Challenge was publicized on mailing lists, at conferences and through private communication. It has been held in conjunction with several major conferences in the area. It provides a web page with comprehensive information and several mailing lists to accommodate different levels of involvement in the communication about the initiative. Participation in the benchmarking is open to everyone, but the evaluation can be executed offline only in a limited way. However, evaluation events are repeated at least on a yearly basis. There is no fee or other restriction for participating in the benchmarking, however, attendance of a workshop is required for participation. Depending on the venue this has sometimes required a workshop or conference registration fee in addition to the other travel expenses. Results and findings of the evaluation are discussed at open workshops and published in detail on a web page. All information is freely available.

B.2.2. Feasibility Requirements

The Feasibility Requirements are intended to ensure than an evaluation is planned and conducted in a realistic, thoughtful, diplomatic and cost-effective manner.

Feasibility 1 (Appropriate Procedures) *Evaluation procedures, including information collection procedures, shall be chosen so that the burden and cost placed on the stakeholders is appropriate in comparison to the expected benefits of the evaluation.*

The pros and cons of the chosen evaluation setup including alternative procedures have been discussed within a W3C Incubator Group and at several workshops. A testbed supporting the evaluation is available but the execution of the evaluation and analysis of the collected data has not been automated to the greatest possible extent. Automation has been implemented to the extent possible within the limits of the resources of the benchmarking organizers.

Feasibility 2 (Diplomatic Conduct) *The evaluation shall be planned and conducted so that it achieves maximal acceptance by the different stakeholders with regard to the evaluation process and findings.*

The setup of the evaluation campaign explicitly encourages participants to reuse each other results, to form teams and to learn about each other's approaches by preparing jointly authored comparisons of the various technologies. The benchmark is organized as a challenge rather than a contest and there is no declared winner of the evaluation event. Presentation of evaluation results is discussed at each workshop before the certification results are put online.

Feasibility 3 (Evaluation Efficiency) *The relation between cost and benefit of the evaluation shall be appropriate.*

Participants in the evaluation need skills about SOAP based Web services for two of the three problem scenarios. Otherwise, no skills except for their own technologies are required. Unfortunately, no estimates of the minimal or optimal predicted time necessary to solve a problem scenarios or other explicit cost-benefit have been provided so far. Monetary cost is involved for the mandatory participation in an evaluation workshop. Apart from the cost, the benchmarking can be scaled to be more or less complex by attempting to solve more or less problem scenarios.

B.2.3. Propriety Requirements

The Propriety Requirements are intended to ensure that in the course of the evaluation all stakeholders are treated with respect and fairness, that the evaluation achieves maximum objectivity and provides an unbiased and appropriate analysis of the technologies under examination.

Propriety 1 (Formal Agreement) *Obligations of the formal parties to an evaluation (what is to be done, how, by whom, when) shall be agreed to in writing, so that these parties are obligated to adhere to all conditions of the agreement or to renegotiate it.*

No written agreement about the responsibilities and commitments of everyone involved in the evaluation has been prepared.

Propriety 2 (Protection of Individual Rights) *The evaluation shall be designed and conducted in a way that protects the welfare, dignity and rights of all stakeholders.*

Results are discussed at the workshops before they are released. Furthermore, participants are encouraged to vet their results. There is no explicit process or option for users to veto the release of their results. However, so far, this has never been an issue.

Propriety 3 (Complete and Fair Investigation) *The evaluation shall undertake a complete and fair examination and description of strengths and weaknesses of the evaluand so that strengths can be built upon and problem areas addressed.*

The evaluation results provide detailed information about the tools under investigation. Since participants are required to prepare a paper about their entry, they

have the option to give appropriate coverage of the strengths and weaknesses of the evaluated technology. Detailed and meaningful comparisons of the evaluated technologies are furthermore provided by means of jointly written comparison papers. The selection of characteristics of a technology under evaluation is determined by the availability of corresponding problem scenarios and not further justified. Participants can freely choose the scenarios and problem levels they address, thus, weaknesses may become visible only implicitly. Participants are encouraged to submit new problem scenarios if they feel that important characteristics of their technology are not yet covered.

Propriety 4 (Unbiased Conduct and Reporting) *The evaluation shall take into account the different views of the stakeholders concerning the evaluand and the evaluation findings. Like the entire evaluation process, the evaluation report shall evidence the impartial position of the evaluation team. Value judgments shall be made as unemotionally as possible.*

Independence of the evaluation from any particular solution is ensured by encouraging the submission of new problem scenarios and through the formal problem scenario approval process. However, single problem scenarios may not be independent from the technology of the people submitting those scenarios. The evaluation is entirely independent from a particular platform or technology except that two scenarios require the interaction with standard SOAP based Web services. Evaluation scenarios are specified exclusively using natural language English text and standard XML schemata and WSDL files. They are thus applicable to all technologies and not biased in favor of a specific one. The evaluation can be meaningfully applied to research prototypes as well as mature products, even though the usage of the previous may result in additional effort for preparing a working solution. There is a reviewing process for evaluation results and reports (results are discussed at the workshops and participants are encouraged to review the published information).

Propriety 5 (Disclosure of Findings) *As far as possible, all stakeholders shall have access to the evaluation findings.*

Scores and evaluation results are clearly documented. Participants are encouraged to upload their solutions to an FTP server and document it in a way that allows the independent reproducibility of the evaluation findings. However, this process is not mandatory and most solutions have not been documented sufficiently to allow for independent reproducibility of evaluation results. The terms of publication of evaluation results are defined in the calls for participation and the general descrip-

tion of the benchmark. Publication of results has been performed accordingly. All information collected during the benchmarking is publicly available.

B.2.4. Accuracy Requirements

The accuracy requirements are intended to ensure that an evaluation produces and discloses valid, accurate, precise, reliable and useful information and findings pertaining to the evaluation purposes and questions.

Accuracy 1 (Description of the Evaluand) *The evaluand shall be described and documented clearly and accurately so that it can be unequivocally identified.*

The benchmark clearly identifies the tasks to perform and the characteristics of a technology under investigation. The quality criteria being assessed in the evaluation are clearly defined and the tools or techniques that are intended to be evaluated specified. The evaluation requirements and assumptions are provided.

Accuracy 2 (Context Analysis) *The context of the technologies being evaluated shall be examined and analyzed in sufficient detail.*

The context of the technologies being evaluated is not examined by the benchmarking organizers as part of the evaluation. However, participants are encouraged to discuss these and their potential influence on the evaluation findings in the paper and solution documentation accompanying any entry.

Accuracy 3 (Described Purposes and Procedures) *Object, purposes, methodology and procedures of an evaluation, including the applied methods, shall be accurately documented and described so that they can be identified and assessed.*

The measured characteristics and evaluation criteria as well as the tasks to perform and the input data are clearly defined in the benchmark specification. The procedure how the evaluation is executed and how the resulting scores are compiled are documented properly for the more recently added scenario, but corresponding documentation (especially with respect to sufficient and necessary criteria to be certified) could be improved for the other scenarios. This corresponds to efforts of making evaluation criteria more transparent and rigorous compared to the first evaluation workshops where the procedure was more consensus-based and less well-defined than it is now.

Accuracy 4 (Disclosure of Information Sources) *The information sources used in the course of the evaluation shall be documented in appropriate detail so that the reliability and adequacy of the information can be assessed.*

The raw data on which the evaluation findings are based consist primarily of the code review and demonstration performed during the workshop. This "raw data" is so far not documented (e.g., filmed or otherwise protocolled) in detail. Thus, while it is traceable how the raw data was obtained, it is not easy to verify the quality of the process and data retrospectively.

Accuracy 5 (Valid and Reliable Information) *The data collection procedures shall be chosen and developed and then applied in a way that ensures the reliability and validity of the data with regard to answering the evaluation questions. The technical criteria shall be based on the standards of quantitative and qualitative social research.*

The assumptions made by the evaluation scenarios have been approved by community consensus during the scenario approval process. The performance measures reflect discussion within the community for several years. One person applying the evaluation on the same technology would twice get the same result. Different persons applying the evaluation on the same technology would very likely get the same results if their knowledge of the technology is comparable. Results are not expected to be affected by unpredictable environment behavior. Threats to the evaluation's validity are discussed in Section 8.6.4. Results of the evaluation are affected by the quality of tool support and maturity of implementation to the extent that better tool support reduces the effort of participation and may allow addressing more problem scenarios in the same time. It is possible to extend technologies specifically to be able to solve certain problem scenarios. However, this is not a threat to the validity of the evaluation results since the evaluation assesses the functional scope of a framework at the time of evaluation. A code review is performed regularly as part of the evaluation to audit solutions and prevent against cheating.

Accuracy 6 (Systematic Data Review) *The data collected, analyzed and presented in the course of the evaluation shall be systematically examined for possible errors.*

The infrastructure and software supporting the evaluation has been tested and run in the expected way. However, the documentation of the test runs can be improved. All infrastructure supporting the benchmark is open source. There is no formal review process to test the data being assembled but the evaluation is

performed within an open workshop such that all participants in the workshop can verify the evaluation live.

Accuracy 7 (Analysis of Qualitative and Quantitative Information) *Qualitative and quantitative information shall be analyzed in an appropriate, systematic way so that the evaluation questions can be effectively answered.*

The basic procedure of the benchmark is explained in a way that everyone can understand it but success criteria for some scenarios could be specified more clearly. Values and limitations of the method are discussed within a W3C Incubator Group report [PKMS08] and in Section 8.6.4. The scenario problem levels and associated functional challenges have been approved as good indicators of the evaluation criteria (functional scope) by the organizing committee of the SWS Challenge initiative and several workshops. Admittedly however, the evaluation goal is somewhat modest. As discussed in Section 8.6.4, no judgment about the ease of use of a technology is being made. With this respect, fitness for purpose is restricted to assessing the pure functional capabilities in terms of what can be done with a technology, not how easy it is to do it. Except for this limitation, tools or techniques that do not have fitness for purpose can not obtain a good performance score. However, tools or techniques that do have fitness for purpose may obtain a bad performance score if improperly used. This is prevented by letting the developers of a technology use the technology within the evaluation. Thus, the compiled scores represent the capabilities of single technologies fairly and accurately and allow a direct comparison of two technologies.

Accuracy 8 (Justified Conclusions) *The conclusions reached in the evaluation shall be explicitly justified so that the audiences can assess them.*

The benchmark explicitly avoids making conclusions in particular about the superiority or inferiority of participating technologies beyond the pure assessment of proven capabilities. This assessment is justified by the demonstrated solutions.

Accuracy 9 (Meta-Evaluation) *The evaluation shall be documented and archived appropriately so that a meta-evaluation can be undertaken.*

The evaluation is documented, but as discussed above, much of the evaluation is performed as part of demonstrations and code reviews during workshops. These have not been protocolled in detail. Thus, a meta-evaluation is difficult to perform without having participated in the evaluation workshops. There are no alternative evaluations with similar scope or purpose with which this benchmark could have been compared.

APPENDIX C

Additional Information on the SWS Matchmaking Benchmark

This appendix makes some additional information about the benchmark for SWS matchmakers presented in Chapter 7 available for reference. Section C.1 presents an assessment of the benchmark with respect to the evaluation requirements specified in Section 4.2.

C.1. Assessment with Respect to Evaluation Requirements

In this section, an analysis of the SWS Matchmaking Benchmark with respect to the evaluation requirements identified in Section 4.2 is provided. Each requirement is briefly stated and a short statement with respect to the questions operationalizing the requirement is given. For improved legibility, the questions are not repeated but are available in Section 4.2. The analysis presented here served as the basis for the corresponding summary in Section 8.7.1.

C.1.1. Utility Requirements

The Utility Requirements are intended to ensure that an evaluation is guided by both the clarified purposes of the evaluation and the information needs of its intended users.

Utility 1 (Stakeholder Identification) *Persons or groups potentially interested in or affected by the evaluation should be identified and contacted, so that their*

interests can be clarified and taken into consideration when designing the evaluation.

The benchmark has been developed under the umbrella of the S3 Contest initiative. It has also been extensively discussed on the related SWS Challenge mailing lists. There were several public calls for feedback during the development of the benchmark. Selected groups in the community have also been contacted directly. Finally, the benchmark setup was discussed at the 6th European Semantic Web Conference.

Utility 2 (Clarification of the Purposes of the Evaluation) *The purposes of the evaluation shall be stated clearly, so that the stakeholders can provide relevant comments on these purposes, and so that the evaluation team knows exactly what it is expected to do.*

The use case, assumptions, scope and purposes of the evaluation are clearly defined in the benchmark description. Improvement of the benchmarked tools is an explicit goal of the evaluation. Evaluation results contain some hints towards improvement recommendations, but a detailed analysis of the evaluation results with this respect is left to the participants.

Utility 3 (Evaluator Credibility and Competence) *The persons conducting an evaluation shall be trustworthy as well as methodologically and professionally competent, so that the evaluation findings achieve maximum credibility and acceptance.*

The evaluation has been organized primarily by the candidate, however, it has been organized under the umbrella of the S3 Contest whose organizing committee represents the diversity of the interested research community well. The benchmark has been developed primarily by one group (although feedback from different groups was collected). The benchmarking campaign is organized by people from seven institutions. People in the wider community were frequently contacted for feedback and asked to become involved actively.

Utility 4 (Information Scope and Selection) *The scope and selection of the collected information shall make it possible to answer relevant questions about the evaluand and, at the same time, consider the information needs of the client and other stakeholders.*

The design of the evaluation was discussed several times, primarily via community mailing lists. There are related evaluation approaches represented by the other tracks of the S3 Contest. The benchmark was designed to extend and complement

these. The problem of SWS discovery and retrieval is found in actual practice, the data being used in the track represents actual data from practice. The selection of input data was justified by a survey of publicly available Web services. Input data is available in four different sizes to accommodate different levels of commitment by the participants. The benchmark task can be scaled well based upon the availability of test data. The task can be solved and poor as well as good solutions are possible. This is also indicated by the performance variance encountered in the evaluation results.

Utility 5 (Transparency of Values) *The perspectives and assumptions of the stakeholders that serve as a basis for the evaluation and the interpretation of the evaluation findings shall be described in a way that clarifies their underlying values.*

The assumptions of the evaluation use case scenario are described in detail. Participants were asked to provide information about their technologies. This information was included in the evaluation result report.

Utility 6 (Report Comprehensiveness and Clarity) *Evaluation reports shall provide all relevant information and be easily comprehensible.*

The setup of the evaluation including prerequisites, assumptions, input data, roles and tasks, information collection and data analysis is clearly defined and described. The findings of the evaluation are considered to be completely comprehensible for all stakeholders and also for interested outsiders.

Utility 7 (Evaluation Timeliness) *The evaluation shall be initiated and completed in a timely fashion so that its findings can inform pending decision and improvement processes.*

The benchmark is intended to be used repeatedly. Its execution is embedded in a yearly series of benchmarking events. The event is primarily organized remotely and deadlines are released long before the event. Participants were repeatedly asked to provide feedback to improve the evaluation. The benchmark has been executed once and not yet updated. However, it was designed to be extended and updated and repeated frequently.

Utility 8 (Evaluation Utilization and Use) *The evaluation shall be planned, conducted, and reported in ways that encourage attentive follow-through by stakeholders and utilization of the evaluation findings.*

Extensive efforts to publicize and promote the project were undertaken. The benchmark has been announced several times via public calls for participation, and personal contact to people in the community expected to be potentially interested. It has been presented at the 8th International Semantic Web Conference (ISWC09) and as part of a tutorial on Semantic Web technology evaluation at the 6th European Semantic Web Conference (ESWC09). A public web page is available with comprehensive information about the benchmark. Participation in the evaluation was open to all interested parties. There were no fees involved in participation and the evaluation can be repeated offline any time. All necessary data and software is available. However, relevance judgments are available only upon request (to prevent people from optimizing their system towards the benchmark). Furthermore, the SME2 evaluation environment is publicly available, but not open source. The results of the evaluation were well publicized and discussed at an open workshop at ISWC09. The evaluation results including all raw data are available online without any access limitations.

C.1.2. Feasibility Requirements

The Feasibility Requirements are intended to ensure than an evaluation is planned and conducted in a realistic, thoughtful, diplomatic and cost-effective manner.

Feasibility 1 (Appropriate Procedures) *Evaluation procedures, including information collection procedures, shall be chosen so that the burden and cost placed on the stakeholders is appropriate in comparison to the expected benefits of the evaluation.*

The evaluation setup including pros and cons and alternative procedures are extensively discussed. Tools have been leveraged to relieve participants of avoidable effort as much as possible. The execution of the evaluation is fully automated using the SME2 environment. However, some analysis of the collected data needs to be performed manually since SME2 does not provide all required functionality.

Feasibility 2 (Diplomatic Conduct) *The evaluation shall be planned and conducted so that it achieves maximal acceptance by the different stakeholders with regard to the evaluation process and findings.*

The benchmark was designed such that participants are encouraged to learn from each others approaches. There is no awarded winner to this benchmark and the corresponding track even though the S3 Contest usually names winners of each edition. The presentation of evaluation results was extensively discussed with participants and participants were asked for approval of results before these were made public.

Feasibility 3 (Evaluation Efficiency) *The relation between cost and benefit of the evaluation shall be appropriate.*

The participants did not need any special skills or training except for their own technologies. The estimated time for participation is not discussed as part of the benchmark. There is no mandatory cost involved in the participation in or the organization of the evaluation. However, participation in the workshop that discusses results is encouraged. This implies corresponding travel and registration fees. The evaluation can be scaled to be more or less complex and costly by using smaller or larger test collections.

C.1.3. Propriety Requirements

The Propriety Requirements are intended to ensure that in the course of the evaluation all stakeholders are treated with respect and fairness, that the evaluation achieves maximum objectivity and provides an unbiased and appropriate analysis of the technologies under examination.

Propriety 1 (Formal Agreement) *Obligations of the formal parties to an evaluation (what is to be done, how, by whom, when) shall be agreed to in writing, so that these parties are obligated to adhere to all conditions of the agreement or to renegotiate it.*

No written agreement about the responsibilities and commitments of everyone involved has been prepared, but the process including roles and tasks has been described in detail and informally agreed upon by all involved people.

Propriety 2 (Protection of Individual Rights) *The evaluation shall be designed and conducted in a way that protects the welfare, dignity and rights of all stakeholders.*

Participants were asked to vet and correct their results. No results were made public without prior written approval by the participants.

Propriety 3 (Complete and Fair Investigation) *The evaluation shall undertake a complete and fair examination and description of strengths and weaknesses of the evaluand so that strengths can be built upon and problem areas addressed.*

The evaluation results provide detailed information about the performance of the tools or methods under investigation and discuss strengths and weaknesses of the evaluated technologies. They allow for detailed and meaningful comparison

of the evaluated technologies. The potentially interesting characteristics of the technologies are comprehensively covered except for limitations with respect to the available test data. Technologies may behave differently for test data with other characteristics. The selection of test data was justified and furthermore explicitly meant to serve as a starting point towards further evaluations based on extended test data.

Propriety 4 (Unbiased Conduct and Reporting) *The evaluation shall take into account the different views of the stakeholders concerning the evaluand and the evaluation findings. Like the entire evaluation process, the evaluation report shall evidence the impartial position of the evaluation team. Value judgments shall be made as unemotionally as possible.*

The evaluation was entirely independent from particular solutions and not tied to a particular platform and technology. The evaluation tasks and all data were specified at a level of abstraction that ensured its applicability to different technologies without being biased towards specific ones. The benchmark can be used for all technologies in the field under investigation and can be applied to research prototypes as well as mature products. The evaluation provided inputs at different size and queries of different complexity. A reviewing process for the evaluation results and report was in place. The evaluation report discusses characteristics of the used test data that may be perceived as potential biases.

Propriety 5 (Disclosure of Findings) *As far as possible, all stakeholders shall have access to the evaluation findings.*

All scores and evaluation results are clearly documented online. All raw data is available such that stakeholders can comprehend and reproduce all evaluation findings. However, disclosure of evaluation findings has not been documented formally in written form at the beginning of the evaluation.

C.1.4. Accuracy Requirements

The accuracy requirements are intended to ensure that an evaluation produces and discloses valid, accurate, precise, reliable and useful information and findings pertaining to the evaluation purposes and questions.

Accuracy 1 (Description of the Evaluand) *The evaluand shall be described and documented clearly and accurately so that it can be unequivocally identified.*

The benchmark defines clearly what is under investigation, which quality criteria are being assessed, which tools or techniques are intended to be evaluated and what are the requirements and assumptions of the evaluation.

Accuracy 2 (Context Analysis) *The context of the technologies being evaluated shall be examined and analyzed in sufficient detail.*

The contexts of the evaluated technologies are described in the evaluation report based upon information collected from participants. The potential influence of the contexts of the evaluated technologies on the evaluation results is discussed, albeit only briefly.

Accuracy 3 (Described Purposes and Procedures) *Object, purposes, methodology and procedures of an evaluation, including the applied methods, shall be accurately documented and described so that they can be identified and assessed.*

The measured characteristics of the systems under evaluation, the evaluation criteria, the tasks to perform and the input data and the procedure how the evaluation is executed, how the resulting scores are compiled and how these are to be interpreted are clearly defined and described in detail. The infrastructure and software supporting the data collection and analysis is properly documented even though the software could be improved with some respects.

Accuracy 4 (Disclosure of Information Sources) *The information sources used in the course of the evaluation shall be documented in appropriate detail so that the reliability and adequacy of the information can be assessed.*

All raw data on which the evaluation findings are based is accessible. It is clearly traceable how the raw data was obtained.

Accuracy 5 (Valid and Reliable Information) *The data collection procedures shall be chosen and developed and then applied in a way that ensures the reliability and validity of the data with regard to answering the evaluation questions. The technical criteria shall be based on the standards of quantitative and qualitative social research.*

The assumptions made by the evaluation about the expected user, the usage context etc. are considered to be realistic. The selection of performance measures is justified and supported by empirical work on the sensitivity and reliability of these measures. Different persons applying the evaluation on the same technology twice

would get the same results, except for the influence that different semantic descriptions have on the evaluation results. This influence is easily traceable and can be isolated if different sets of descriptions are available. The evaluation results are not affected by any unpredictable environment behaviors. Threats to the evaluation's validity may result from characteristics of the test data used and are clearly identified and discussed. Technologies can not be optimized for the measures being used, but they can be optimized for the test data being used. There is no auditing procedure in place to prevent against cheating. However, cheating was made difficult by the evaluation setup.

Accuracy 6 (Systematic Data Review) *The data collected, analyzed and presented in the course of the evaluation shall be systematically examined for possible errors.*

The infrastructure and software supporting the evaluation has been tested, but the SME2 environment had to be extended on relatively short notice. Furthermore, there have been a few bugs in this software. However, these bugs are not expected to affect the evaluation results. The source code of the SME2 environment is not publicly available. All data assembled during the evaluation was given to participants to allow them reviewing the data's correctness.

Accuracy 7 (Analysis of Qualitative and Quantitative Information) *Qualitative and quantitative information shall be analyzed in an appropriate, systematic way so that the evaluation questions can be effectively answered.*

The benchmark and all associated measures have been documented and explained such that everyone can understand them. Values and limitations of the methods used are discussed. The compiled measures used in the evaluation are considered to be good indicators of the performance of the technologies with respect to the quality criteria of interest. A tool that does not have fitness for purpose can not obtain a good performance score. A tool that has fitness for purpose may obtain a bad performance score if the descriptions created for the tool are inappropriate. This is prevented by letting the developers of the tool create these descriptions. The scores compiled by the benchmark represent the capabilities of the evaluated technologies fairly and accurately with respect to the available test data. The scores allow direct comparison of the evaluated technologies.

Accuracy 8 (Justified Conclusions) *The conclusions reached in the evaluation shall be explicitly justified so that the audiences can assess them.*

Conclusions are drawn very cautiously and primarily left to the participants. The scope and limitations of the conclusions are stated clearly and primarily subject to restrictions of the test data being used.

Accuracy 9 (Meta-Evaluation) *The evaluation shall be documented and archived appropriately so that a meta-evaluation can be undertaken.*

The key purposes, steps, methods, data and findings of the evaluation have been comprehensively documented and archived. Comparison with previous or alternative evaluations has been performed in the motivation of the benchmark.